Как мы Change Data Capture делали

Александр Деулин
Василий Тюбек







Tarantool на Highload





Деулин Александр

aleksandr.deulin@nexign.com



Тюбек Василий

v.tyubek@corp.mail.ru



В предыдущем сезоне “Микросервисов”...



Применение микросервисов в высоконагруженном биллинге

https://www.highload.ru/siberia/2019/abstracts/5384



Все началось с кэшей

	Channels				Partners		
Channels	Web/Mobile App	Call Center	...	Web Shop & Retail	Social NW	...	Delivery

Digital ecosystem	**Middleware Technical Enablers**	SSO	AP Gateway	Service Mesh	Container Orchestration	Telemetry Log/ Mon/Trace	API Portal Repository	CI/CD DevOps Tools
	Transactional Caches	**Customer Cache**	**Product Instances Cache**	**Location Cache**	**Interaction Cache**	**Notification Cache**	
	Business Enablers Microservices	Customer 360^0 View: Lead Mgmt; Customer Topology; Ext. Attributes & Personalization	Federated Product Catalogue: Presentation in Channels; Partner Federation; Personal Offers	Ordering: Basket; Check In; CPQ; Orchestration	Customer Interaction: Channel Aggregator; Flow Controller; Predictive Analytics	Notification Hub: Notification Temp. & Policy; Event Processing; Notification Engine; Channel Adapter	Real Time Marketing: …	Partner Integration Adapters: …

Aggregation Layer	**Aggregation Layer**	Complex Event Processing RT	Change Data Capture (Kafka, RMQ, GG)	Shared Meta Model MDM	Bulk Read/Write ETL

Core Services	**Core Service Domains**	BSS	OSS	ERP	DWH	…	Partner Domains



Вначале было просто

SELECT * FROM TABLE

ORACLE

ORACLE

ORACLE

TARANTOOL

TARANTOOL

TARANTOOL

RabbitMQ

CSV



Халявы не будет

Витрина продуктов абонента:

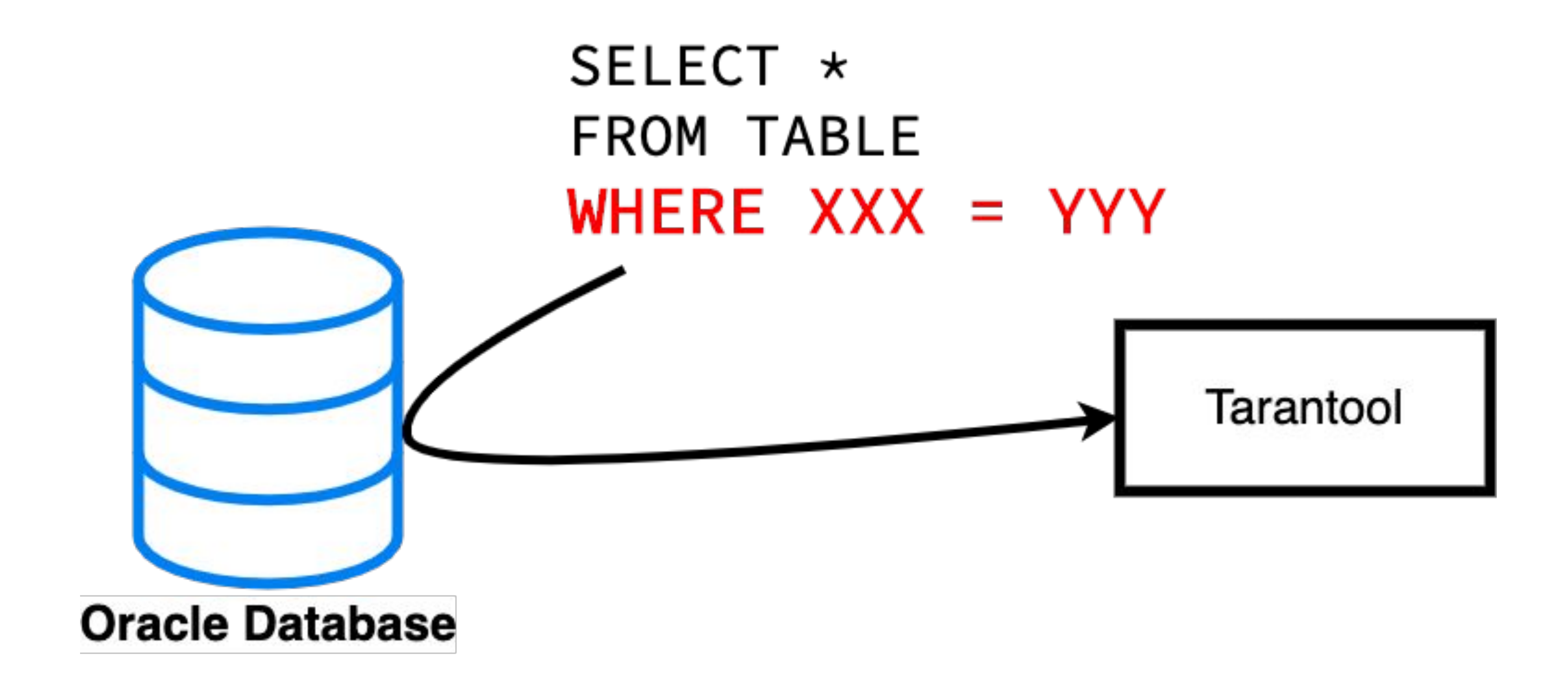
- ~30Tb сырых данных в Oracle, ~3000 TPS обновлений
- Нет брокера, из которого можно прочитать эти изменения

- Данные в витрине должны быть транзакционно целостные
- Отставание витрины — не более 30 секунд от Oracle

- **~1Тб данных в Tarantool, до 1Tb изменений в сутки**



Прогрев


SELECT *
FROM TABLE
WHERE XXX = YYY
Tarantool
Oracle Database


HL
HighLoad++
Весна 2021

Халявы не будет. Совсем

- при прогреве использовать **WHERE** нельзя
  - нагрузили Oracle
  - время выгрузки **~2 недели**
- данные на момент выгрузки уже неактуальные
- ... и это мы еще до получения изменений не дошли



Халявы не будет. Окончательно

- Replica
- CQN
- GoldenGate



GoldenGate

- Транзакционная целостность
- Полноценная поддержка ETL
- Минимальная нагрузка на базу
- Экспертиза DBA



CC*

CS*

CA*

BI*

SF*

Oracle GoldenGate

Customer Data Integration

Product Inventory Cache

Customer Search Index

Kafka Integration



Как делали

- Первый подход к шарду:
  - Прогрев через выгрузку/загрузку CSV
  - Репликация через XML
- Второй подход к шарду:
  - Прогрев напрямую из Oracle
  - USEREXIT
- Третий подход?....



Техника



Read through

App

CACHE

ORACLE



Кэш сбоку

App
READ /
WRITE
ORACLE
REPLICATION
CACHE
New
App
READ



Работающий кэш

- **Загружен данными (прогрев)**
- Данные актуальны (репликация)

Работающий кэш

- Загружен данными (прогрев)
- **Данные актуальны (репликация)**



Работающий кэш

- Загружен данными (прогрев)
- Данные актуальны (репликация)
- **Observable/maintainable/****able**



Прогрев с выгрузкой в CSV

Выгрузка из Oracle

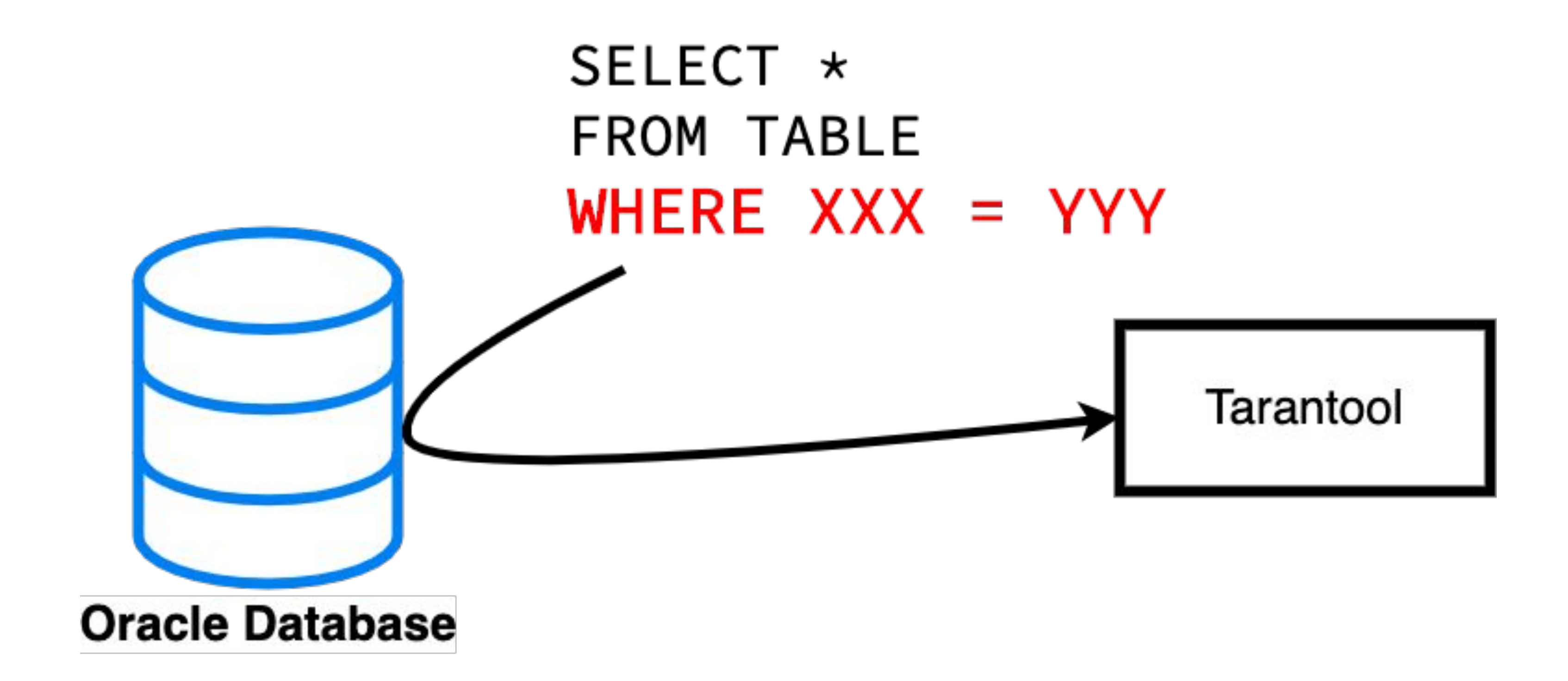


Прогрев

- нам нужен только срез данных
- минимальная нагрузка на Oracle
- адекватное время прогрева



Прогрев


SELECT *
FROM TABLE
WHERE XXX = YYY
Tarantool
Oracle Database


HL
HighLoad++
Весна 2021

“Угнать за 60 секунд”. Минут? Часов?!

WHERE USER IN (1, 1000)

WHERE USER IN (1001, 2000)

WHERE USER IN (2001, 3000)

T

T

T



TABLE

PART 1

PART 2

SubPart 1

SubPart 2

SubPart 1

SubPart 2

T

T

T

T



Прогрев кэша: выгрузка данных

SELECT TABLE_NAME ...
FROM ALL_TABLES

SELECT TABLE_NAME ...
FROM ALL_TAB_PARTITIONS

SELECT TABLE_NAME ...
FROM
ALL_TAB_SUBPARTITIONS

SELECT * FROM PARTITION_1
SELECT * FROM PARTITION_2
...
SELECT * FROM PARTITION_N

Exporter

CSV CSV CSV
CSV CSV CSV

Tarantool Replicator
Tarantool Replicator
Tarantool Replicator

Storage
Storage
Storage



Прогрев кэша: выгрузка данных

SELECT TABLE_NAME ...
FROM ALL_TABLES

SELECT TABLE_NAME ...
FROM ALL_TAB_PARTITIONS

SELECT TABLE_NAME ...
FROM
ALL_TAB_SUBPARTITIONS

SELECT * FROM PARTITION_1

SELECT * FROM PARTITION_2

...

SELECT * FROM PARTITION_N

Exporter

CSV CSV CSV
CSV CSV CSV

CSV CSV CSV

CSV CSV

CSV

Tarantool Replicator

Tarantool Replicator

Tarantool Replicator

Storage

Storage

Storage



Прогрев кэша: выгрузка данных

SELECT TABLE_NAME ...
FROM ALL_TABLES

SELECT TABLE_NAME ...
FROM ALL_TAB_PARTITIONS

SELECT TABLE_NAME ...
FROM
ALL_TAB_SUBPARTITIONS

SELECT * FROM PARTITION_1
SELECT * FROM PARTITION_2
...
SELECT * FROM PARTITION_N

Exporter

CSV CSV CSV
CSV CSV CSV

CSV CSV CSV
CSV CSV
CSV

Tarantool Replicator
Tarantool Replicator
Tarantool Replicator

Storage
Storage
Storage



Прогрев кэша: выгрузка данных

SELECT TABLE_NAME ...
FROM ALL_TABLES

SELECT TABLE_NAME ...
FROM ALL_TAB_PARTITIONS

SELECT TABLE_NAME ...
FROM
ALL_TAB_SUBPARTITIONS

SELECT * FROM PARTITION_1
SELECT * FROM PARTITION_2
...
SELECT * FROM PARTITION_N

Exporter

CSV CSV CSV
CSV CSV CSV

CSV CSV CSV → Tarantool Replicator
CSV CSV → Tarantool Replicator
CSV → Tarantool Replicator

Storage
Storage
Storage



Особенности выгрузки из Oracle

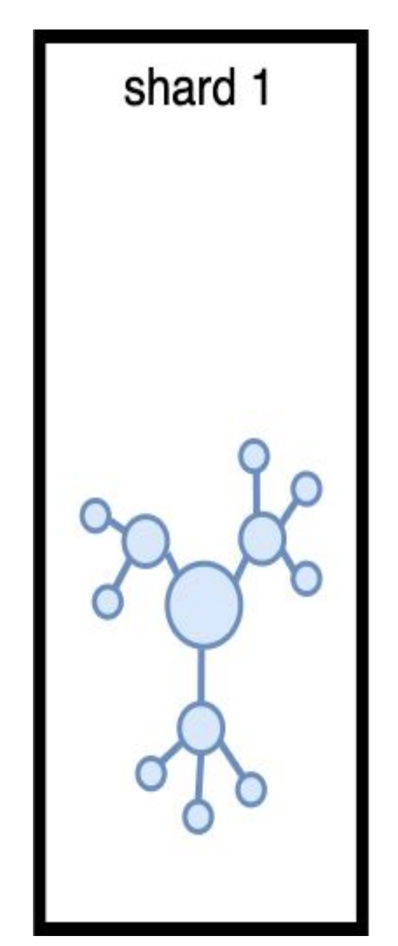
- Никаких WHERE
- Адекватное партиционирование
- prefetch = 1000 (bulk)
- 10, максимум 30 потоков выгрузки
- Время выгрузки — до **6 часов**
- Скорость выгрузки — **60000 - 700 000** строк в секунду
- **1.2Tb** CSV-файлов

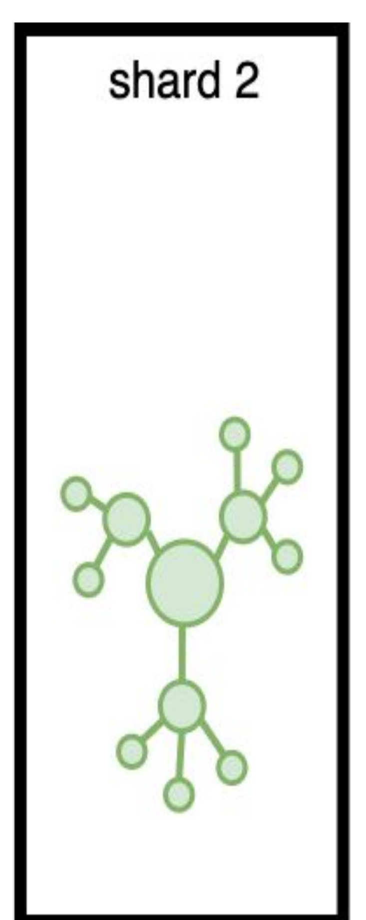


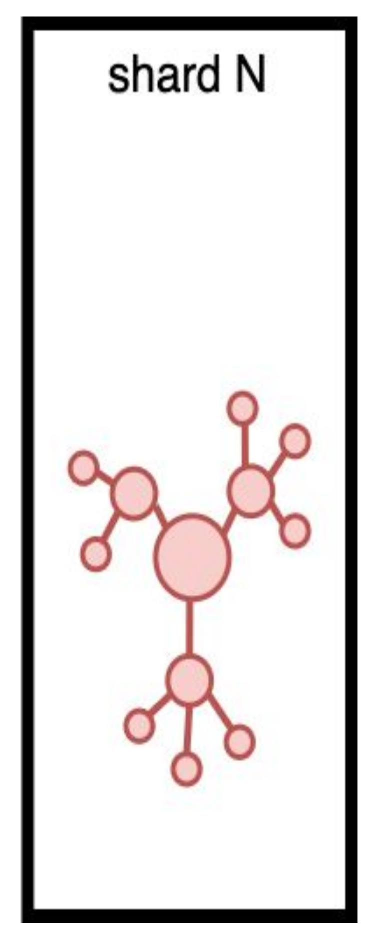
Загрузка в витрину



Модель хранения

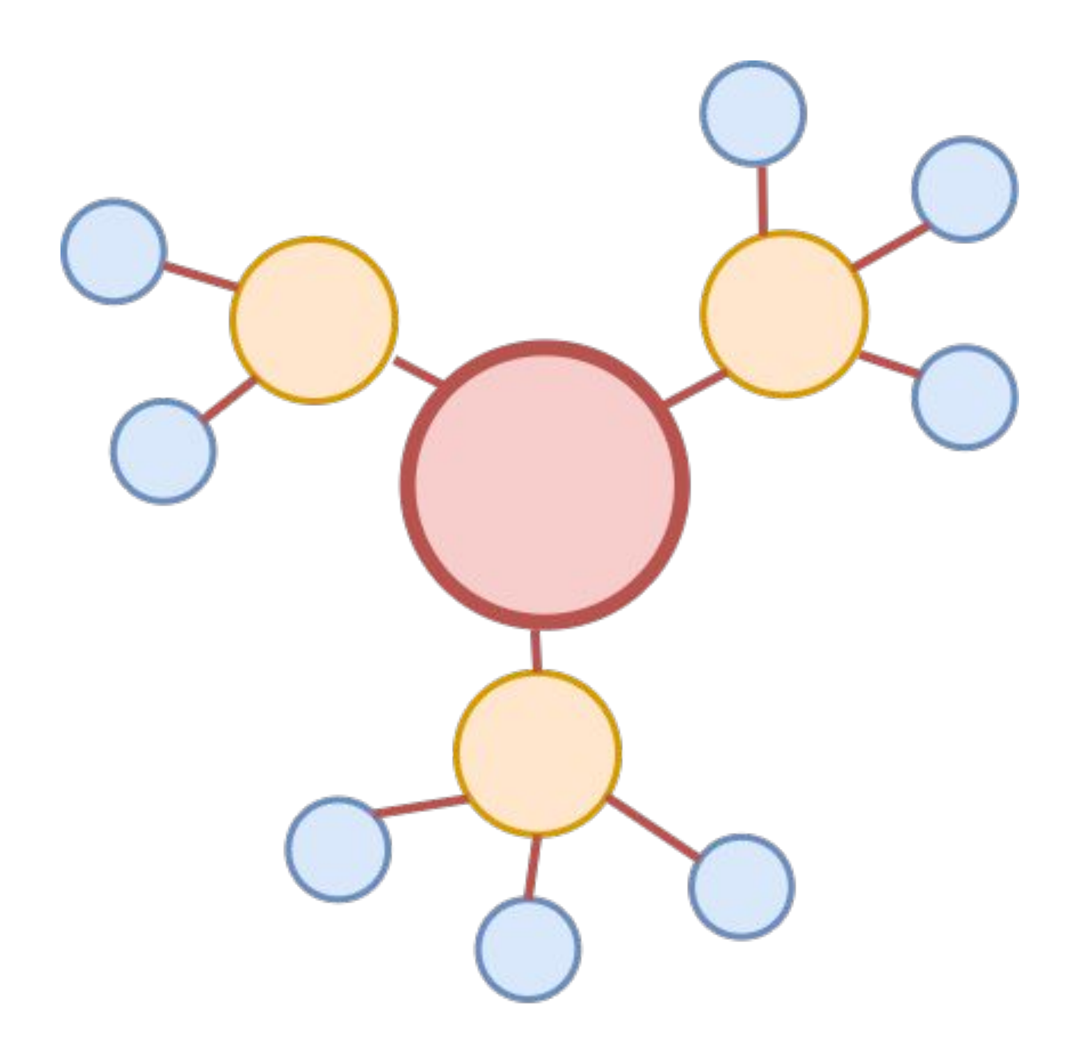






- **Данные шардированы по абоненту**
- У абонента есть подключенные услуги (~4 млрд записей)
- Для каждой услуги есть дополнительные данные (еще ~12 млрд записей)

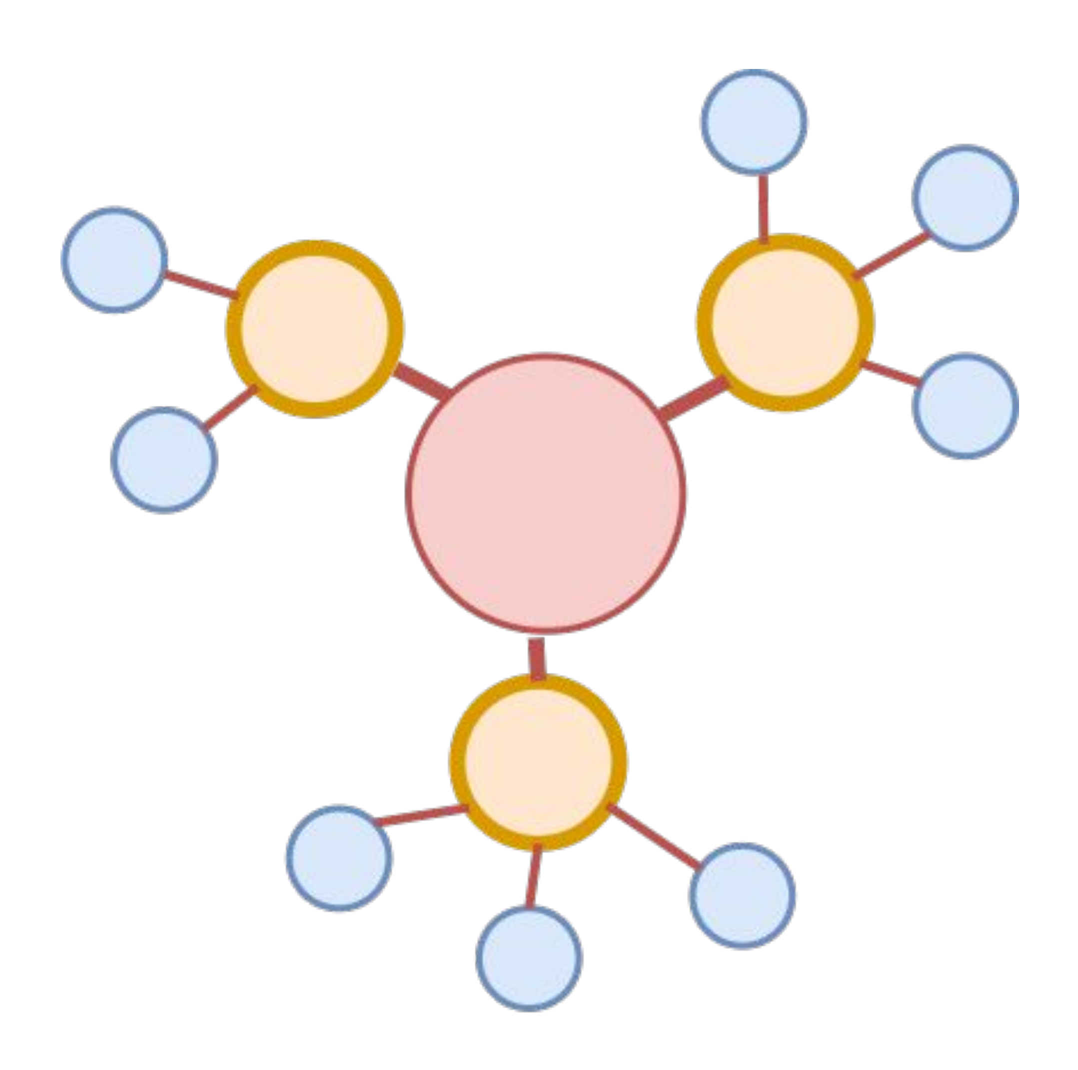
Модель данных



- **Данные шардированы по абоненту**
- У абонента есть подключенные услуги (~4 млрд записей)
- Для каждой услуги есть дополнительные данные (еще ~12 млрд записей)



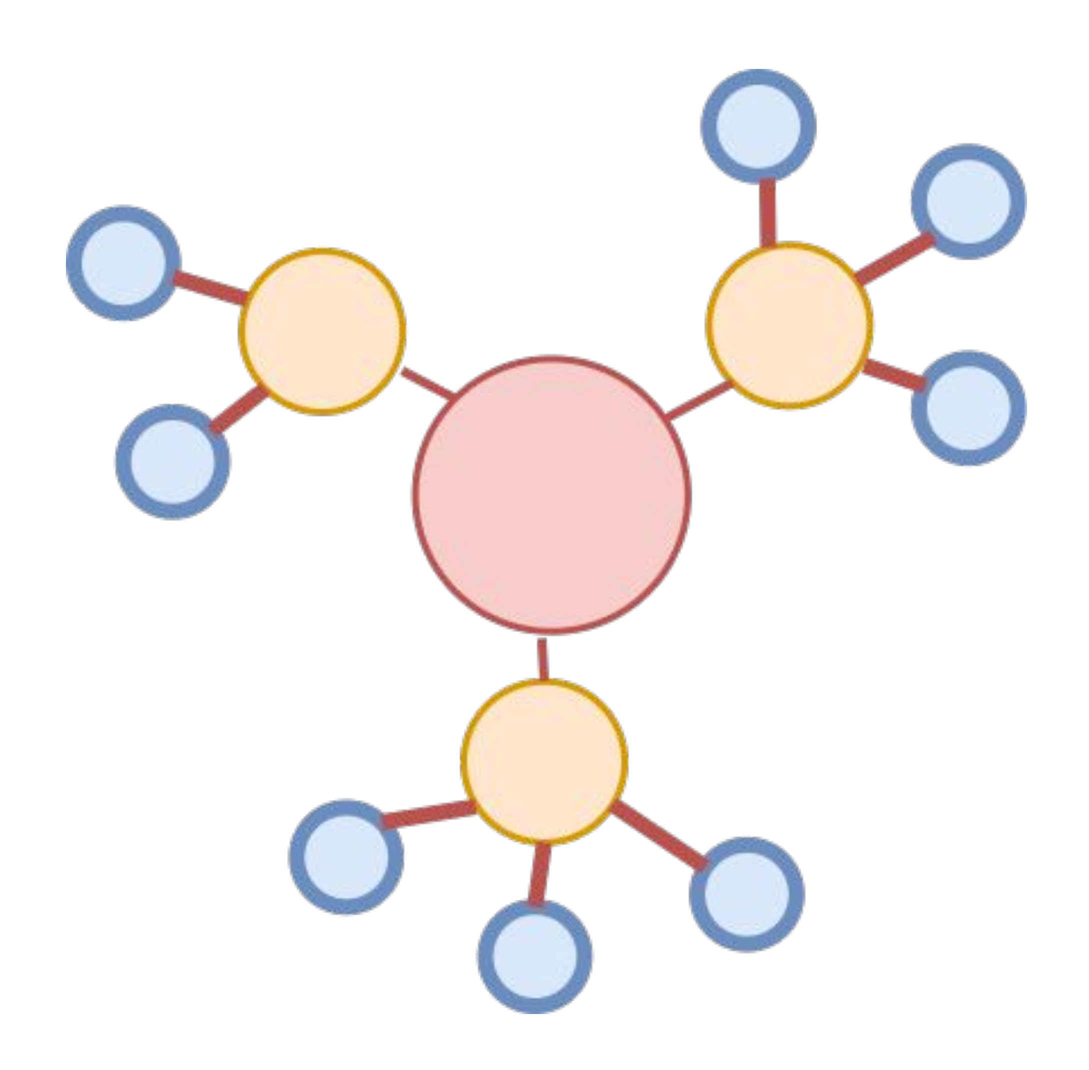
Модель данных



- Данные шардированы по абоненту
- **У абонента есть подключенные услуги (~4 млрд записей)**
- Для каждой услуги есть дополнительные данные (еще ~12 млрд записей)



Модель данных



- Данные шардированы по абоненту
- У абонента есть подключенные услуги (~4 млрд записей)
- **Для каждой услуги есть дополнительные данные (еще ~12 млрд записей)**



Запись данных в кэш

INSERT

shard 1

shard 2

shard N

FIND + INSERT

shard 1

shard 2

shard N



Запись данных в кэш

INSERT

shard 1

shard 2

shard N

FIND + INSERT

shard 1

shard 2

shard N



Прогрев кэша: загрузка данных

SELECT TABLE_NAME ...
FROM ALL_TABLES

SELECT TABLE_NAME ...
FROM ALL_TAB_PARTITIONS

SELECT TABLE_NAME ...
FROM
ALL_TAB_SUBPARTITIONS

SELECT * FROM PARTITION_1
SELECT * FROM PARTITION_2
...
SELECT * FROM PARTITION_N

Exporter

CSV CSV CSV
CSV CSV CSV

Tarantool Replicator
Tarantool Replicator
Tarantool Replicator

Storage
Storage
Storage



Прогрев кэша: порядок загрузки

SELECT TABLE_NAME ...
FROM ALL_TABLES

SELECT TABLE_NAME ...
FROM ALL_TAB_PARTITIONS

SELECT TABLE_NAME ...
FROM
ALL_TAB_SUBPARTITIONS

SELECT * FROM PARTITION_1

SELECT * FROM PARTITION_2

...

SELECT * FROM PARTITION_N

Exporter

CSV CSV CSV

CSV CSV CSV

Tarantool Replicator

Tarantool Replicator

Tarantool Replicator

Storage

Storage

Storage



Особенности загрузки

- 1 шаг — грузим данные по абоненту
- 2 шаг — догружаем все привязанные к нему сущности
- отфильтровали данные на загрузке в кэш
- ~12 часов на загрузку данных



Change Data Capture

Oracle GoldenGate



GoldenGate

GoldenGate

Source DB → Extract → Trail Files → Replicat → Target DB



GoldenGate: различные топологии потоков данных

Consolidation

GoldenGate

Duplication

GoldenGate



GoldenGate: фильтрация и трансформация

Filter:

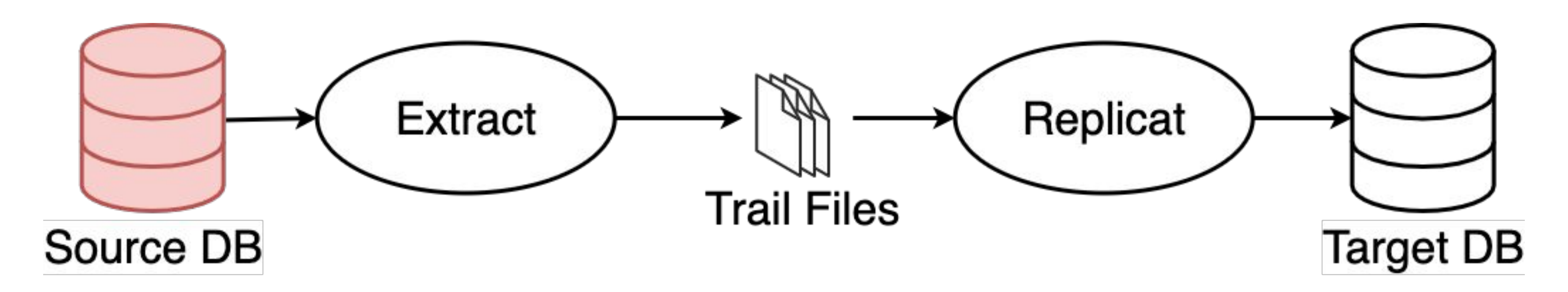
GoldenGate

Transform:

GoldenGate

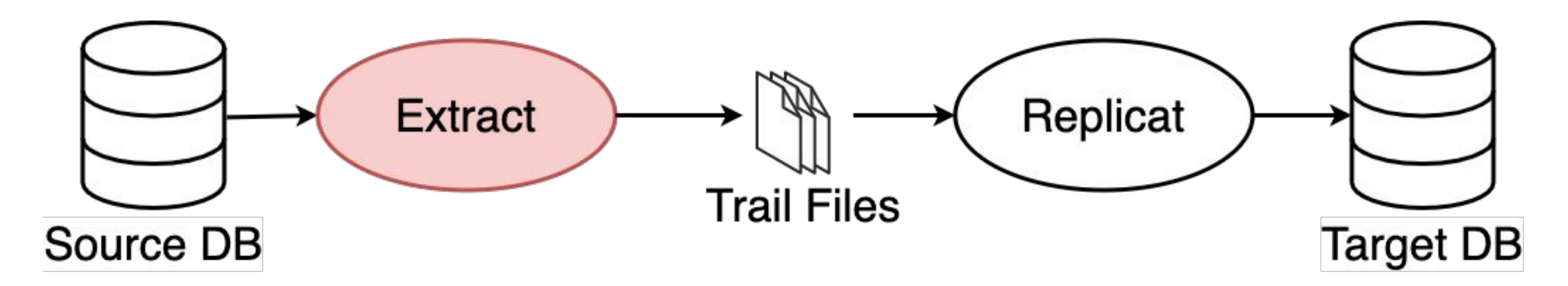


GoldenGate: как работает



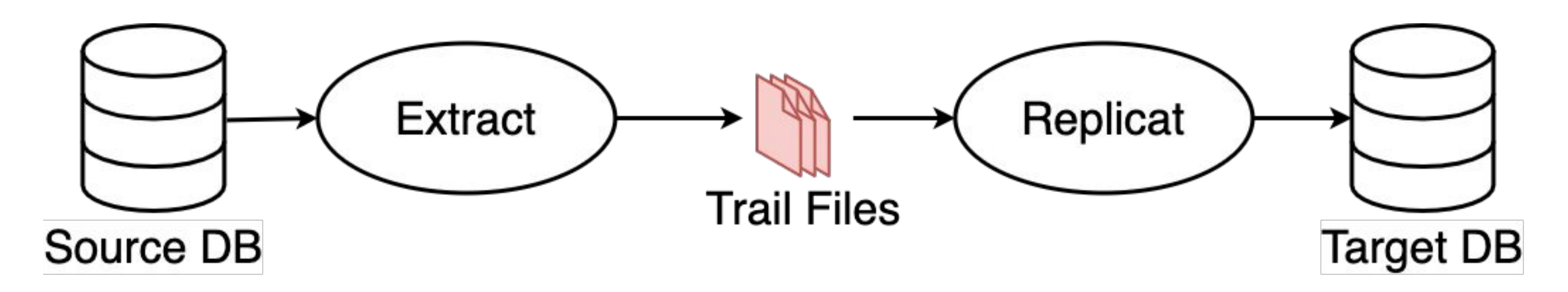


GoldenGate: как работает





GoldenGate: как работает





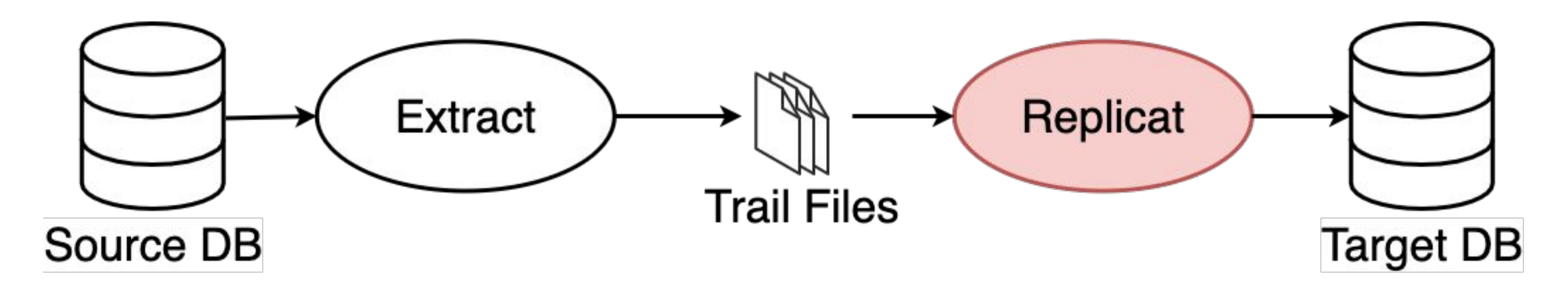
Trail: log транзакций

SCN	Operation
0	T1: insert
10	T2: insert
20	T3: update
30	T4: delete

filename
ds000000000
ds000000001
ds000000002

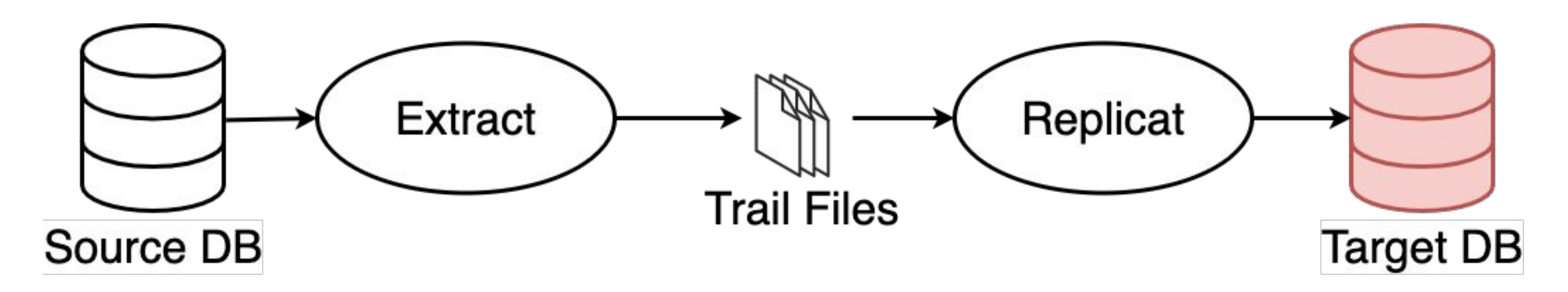


GoldenGate: как работает





GoldenGate: как работает





Summary

- Позволяет реплицировать данные
- Может изменять реплицируемые данные
- Реплицирует только закоммиченные транзакции



Oracle -> Tarantool

XML



GoldenGate: Tarantool

GoldenGate

Source DB

Extract

Trail Files

Replicat

Target DB

GoldenGate

Source DB

Extract

Trail Files

Tarantool Replicator

Tarantool



Trail: форматы

- Бинарный
- ASCI
- SQL
- **XML**



Trail: XML

```
<dbupdate table=”XXX” type=”update” image=”before”>
  <columns>
    <column name=”COLUMN1”>value 1</column>
  </columns>
  <tokens>
    <token name=”X_COMMIT_TIME”>1620802191</token>
  </tokens>
</dbupdate>
```



Trail: XML

```
<dbupdate table=”XXX” type=”update” image=”before”>
  <columns>
    <column name=”COLUMN1”>value 1</column>
  </columns>
  <tokens>
    <token name=”X_COMMIT_TIME”>1620802191</token>
  </tokens>
</dbupdate>
```



Trail: XML

```
<dbupdate table=”XXX” type=”update” image=”before”>
  <columns>
    <column name=”COLUMN1”>value 1</column>
  </columns>
  <tokens>
    <token name=”X_COMMIT_TIME”>1620802191</token>
  </tokens>
</dbupdate>
```

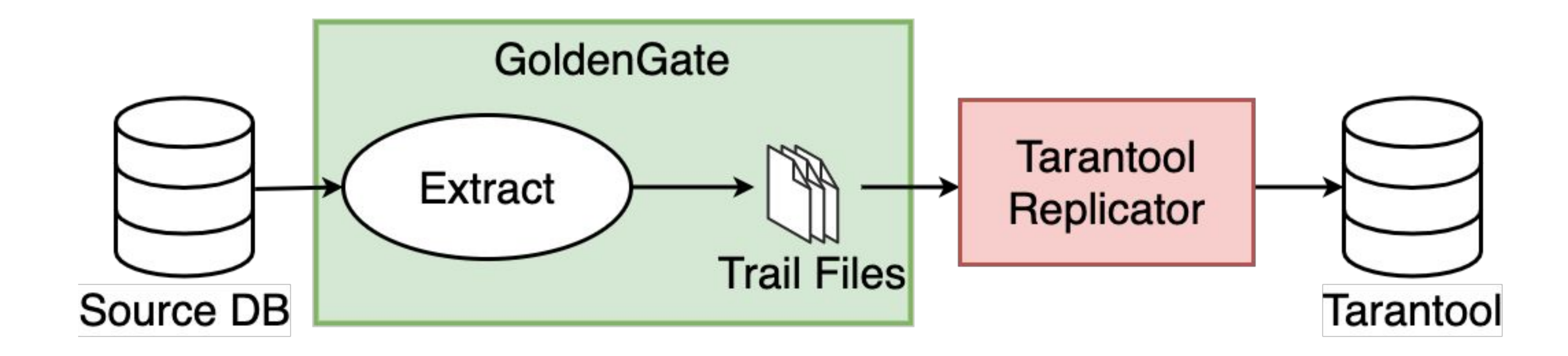


Trail: XML

```
<dbupdate table=”XXX” type=”update” image=”before”>
  <columns>
    <column name=”COLUMN1”>value 1</column>
  </columns>
  <tokens>
    <token name=”X_COMMIT_TIME”>1620802191</token>
  </tokens>
</dbupdate>
```



GoldenGate: Tarantool





Итого

- Грузим данные из CSV
- Репликация из XML
- CSV хорошо на отладке
- XML простой: открыл файл — видишь транзакции



Round 2

FIGHT!



Что можно сделать лучше?



Оптимизируем прогрев

SELECT TABLE_NAME ...
FROM ALL_TABLES

SELECT TABLE_NAME ...
FROM ALL_TAB_PARTITIONS

SELECT TABLE_NAME ...
FROM
ALL_TAB_SUBPARTITIONS

SELECT * FROM PARTITION_1

SELECT * FROM PARTITION_2

...

SELECT * FROM PARTITION_N

Exporter

CSV CSV CSV
CSV CSV CSV

Tarantool Replicator

Tarantool Replicator

Tarantool Replicator

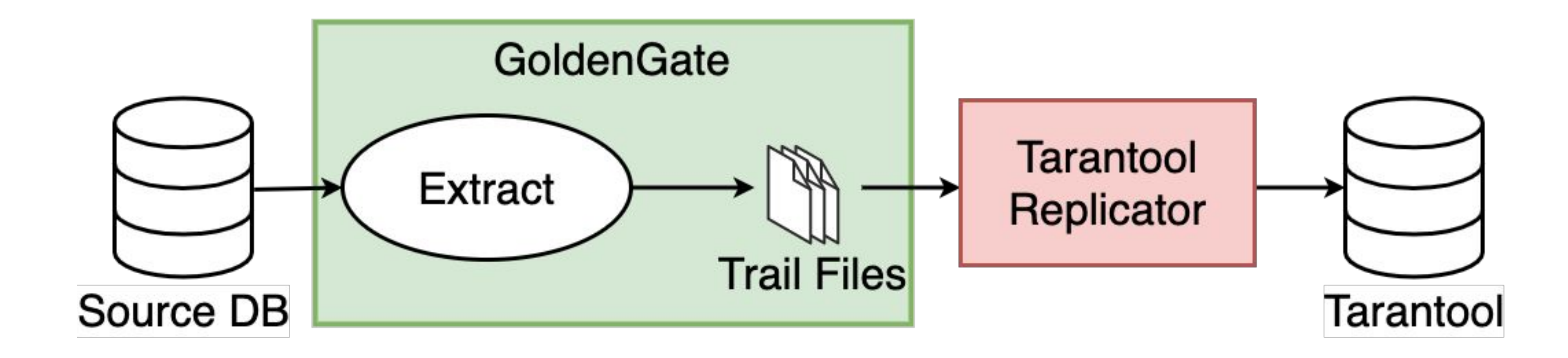
Storage

Storage

Storage



Как не написать свой GG?

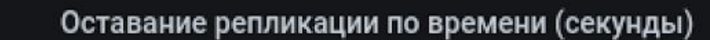




XML: не все типы данных

```
<dbupdate table=”XXX” type=”update” image=”before”>
Unsupported data type CLOB for column COLUMN2
  <columns>
    <column name=”COLUMN1”>value 1</column>
  </columns>
</dbupdate>
```

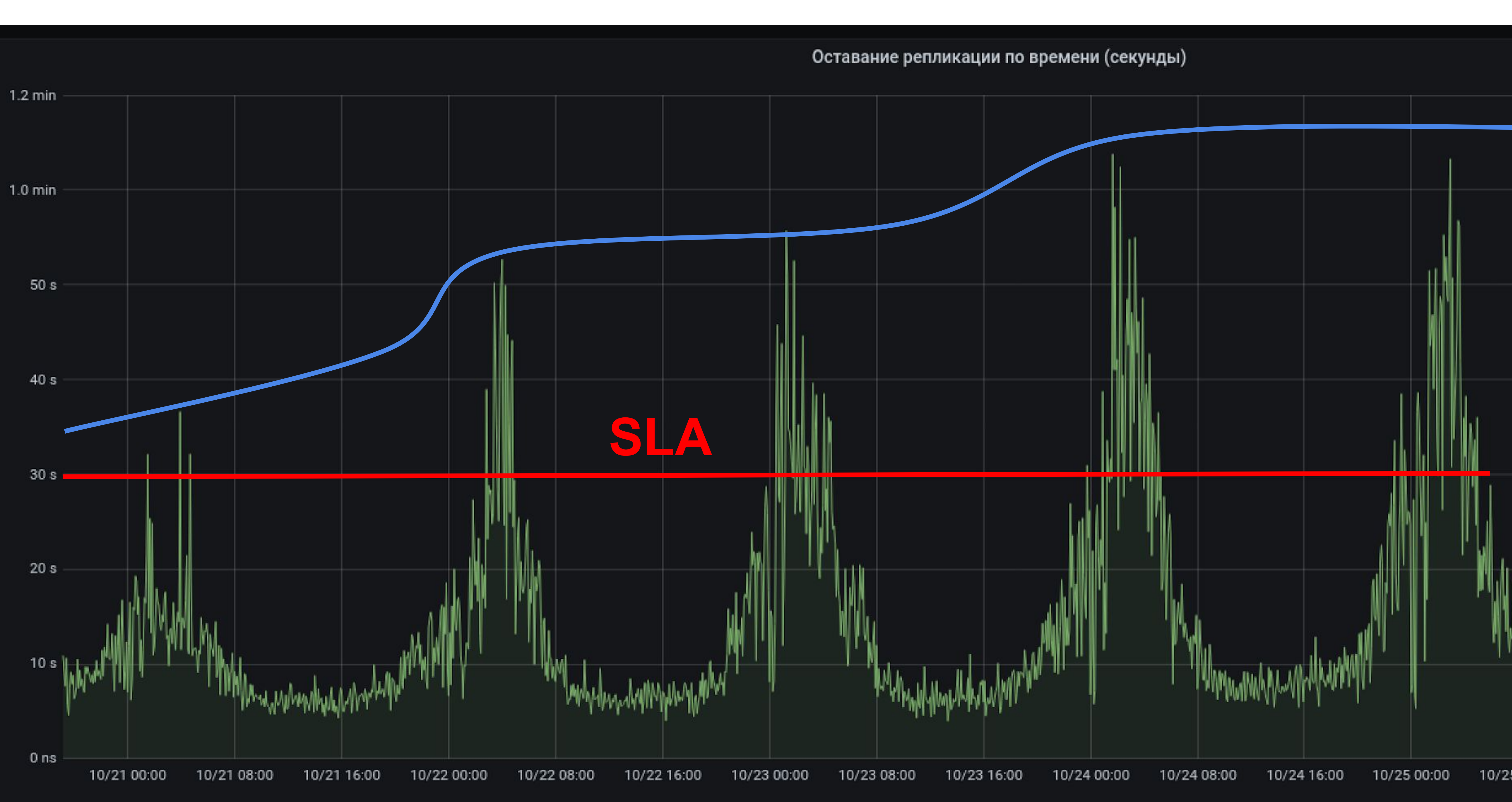




SLA

1.2 min
1.0 min
50 s
40 s
30 s
20 s
10 s
0 ns

10/21 00:00
10/21 08:00
10/21 16:00
10/22 00:00
10/22 08:00
10/22 16:00
10/23 00:00
10/23 08:00
10/23 16:00
10/24 00:00
10/24 08:00
10/24 16:00
10/25 00:00


Оставание репликации по времени (секунды)
1.2 min
1.0 min
50 s
40 s
30 s
20 s
10 s
0 ns
SLA
10/21 00:00
10/21 08:00
10/21 16:00
10/22 00:00
10/22 08:00
10/22 16:00
10/23 00:00
10/23 08:00
10/23 16:00
10/24 00:00
10/24 08:00
10/24 16:00
10/25 00:00

Меньше транзакций -> больше лаг

- транзакции набираем пачками
- не обрабатываем, пока не набрали пачку
- добавили **flush timeout**



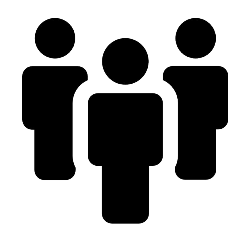
Оставание репликации по времени (секунды)

20 s
15 s
10 s
5 s
0 ns

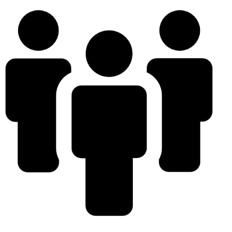
10/27 00:00 10/27 12:00 10/28 00:00 10/28 12:00 10/29 00:00 10/29 12:00 10/30 00:00 10/30 12:00 10/31 00:00 10/31 12:00 11/01 00:00 11/01 12:00 11/02 00:00 11/02 12:00



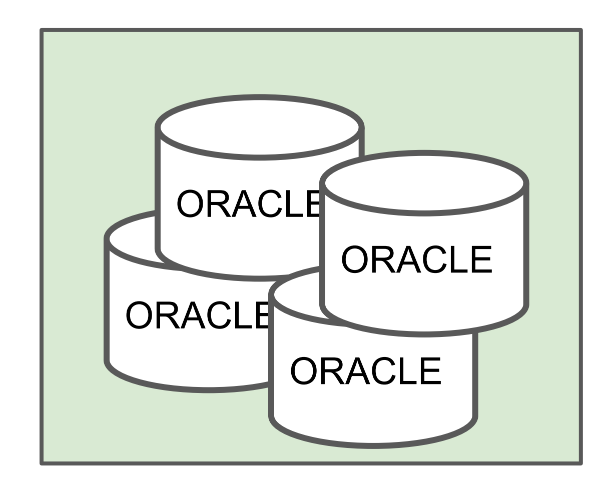
Как не ломать границы эксплуатации?

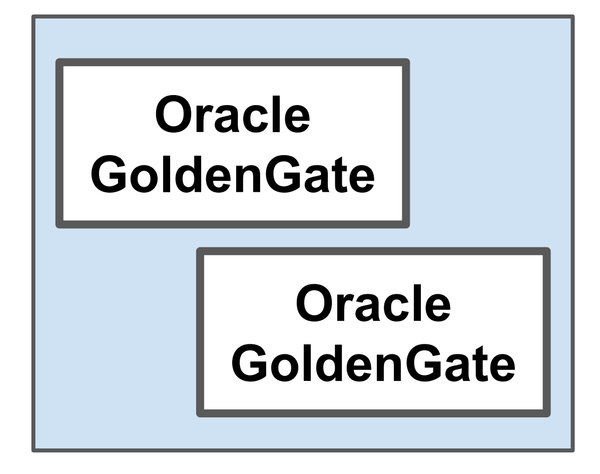


DBA



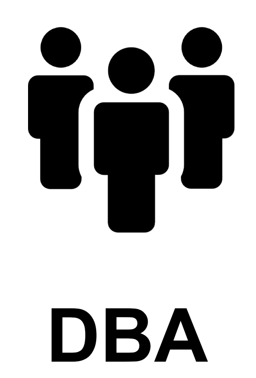
GG

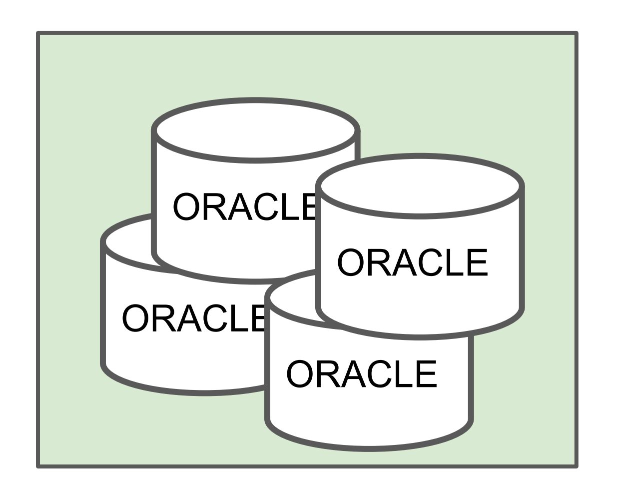


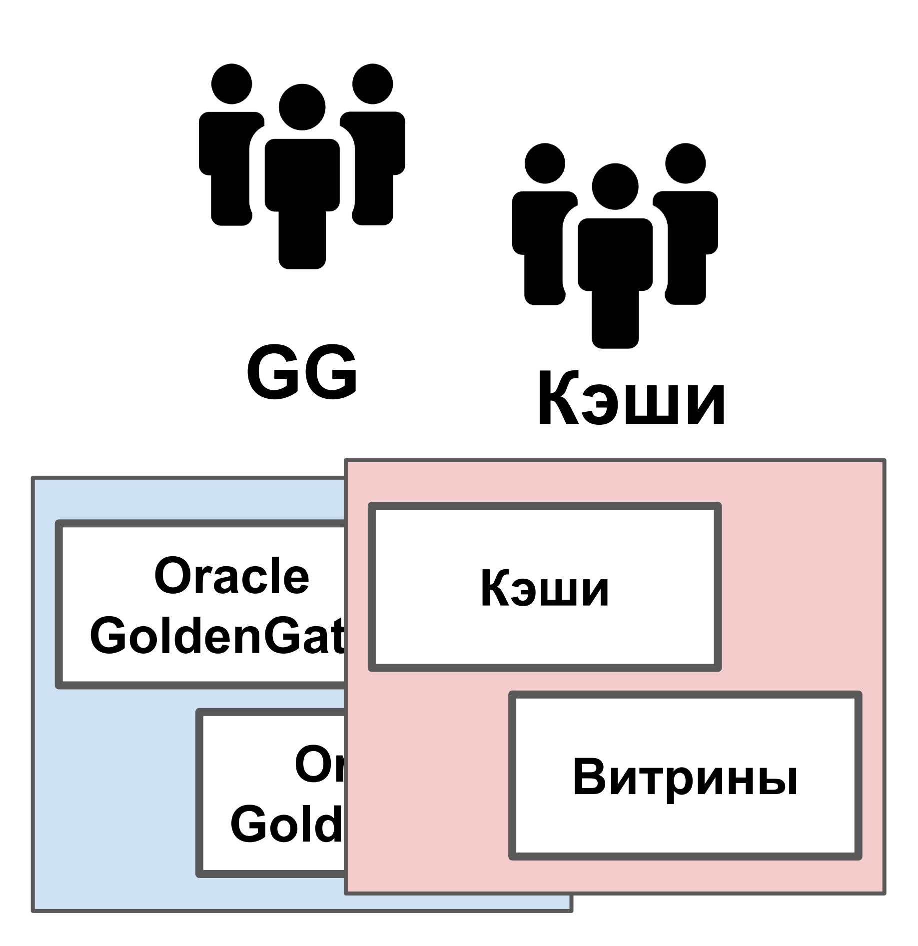




Как не ломать границы эксплуатации?









Прогрев v2



SELECT TABLE_NAME ...
FROM ALL_TABLES

SELECT TABLE_NAME ...
FROM ALL_TAB_PARTITIONS

SELECT TABLE_NAME ...
FROM
ALL_TAB_SUBPARTITIONS

SELECT * FROM PARTITION_1

SELECT * FROM PARTITION_2

...

SELECT * FROM PARTITION_N

SQL SQL SQL

SQL SQL

SQL

Exporter Worker

Exporter Worker

Exporter Worker

Storage

Storage

Storage



SELECT TABLE_NAME ...
FROM ALL_TABLES

SELECT TABLE_NAME ...
FROM ALL_TAB_PARTITIONS

SELECT TABLE_NAME ...
FROM
ALL_TAB_SUBPARTITIONS

SELECT * FROM PARTITION_1

SELECT * FROM PARTITION_2

...

SELECT * FROM PARTITION_N

SQL SQL SQL

SQL SQL

SQL

Exporter Worker

Exporter Worker

Exporter Worker

Storage

Storage

Storage



SELECT TABLE_NAME ...
FROM ALL_TABLES

SELECT TABLE_NAME ...
FROM ALL_TAB_PARTITIONS

SELECT TABLE_NAME ...
FROM
ALL_TAB_SUBPARTITIONS

SELECT * FROM PARTITION_1

SELECT * FROM PARTITION_2

...

SELECT * FROM PARTITION_N

SQL SQL SQL → Exporter Worker

SQL SQL → Exporter Worker

SQL → Exporter Worker

Storage

Storage

Storage



USEREXIT

Change Data Capture



GoldenGate

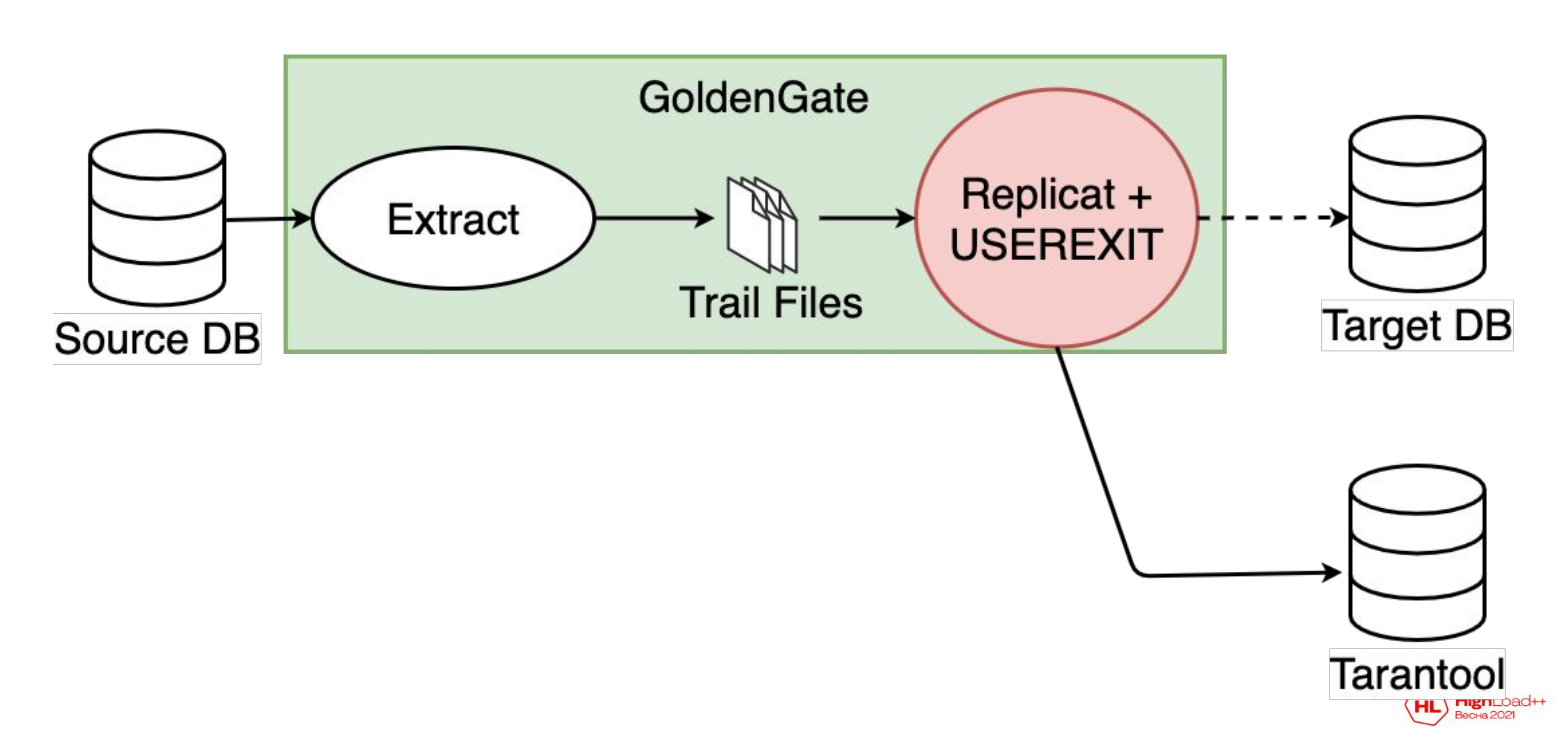
Source DB → Extract → Trail Files → Replicat → Target DB



GoldenGate

Source DB → Extract → Trail Files → Replicat + USEREXIT → Target DB




GoldenGate
Extract
Trail Files
Replicat +
USEREXIT
Source DB
Target DB
Tarantool
HighLoad++
Весна 2021

SLA* < 30s

* на лаг репликации



Оставание репликации по времени (секунды)

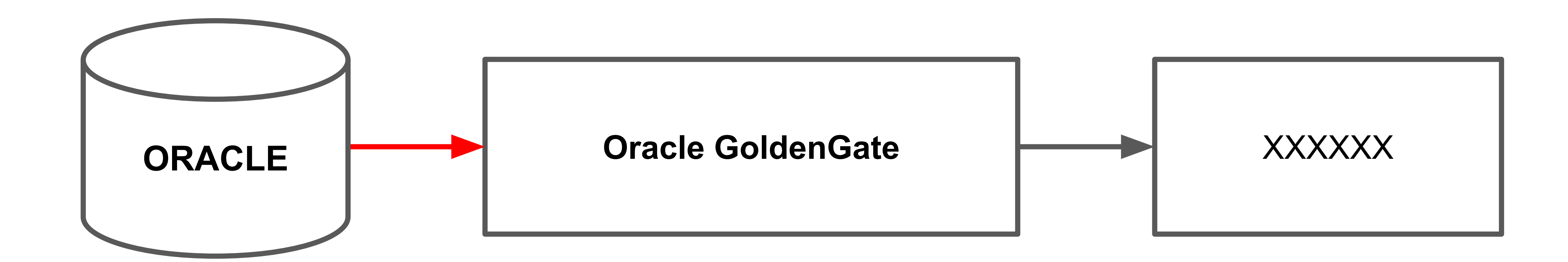
13 min
10 min
7 min
3 min
0 ns

11:20
11:30
11:40
11:50
12:00
12:10

SLA

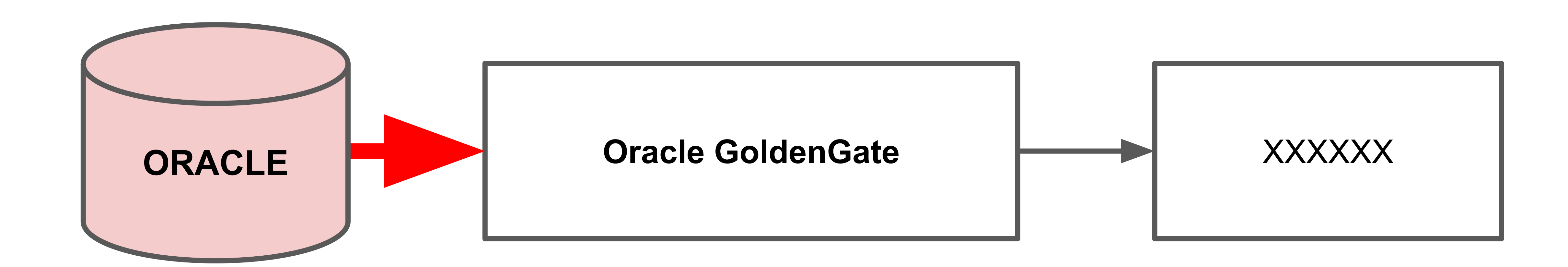


Почему так?



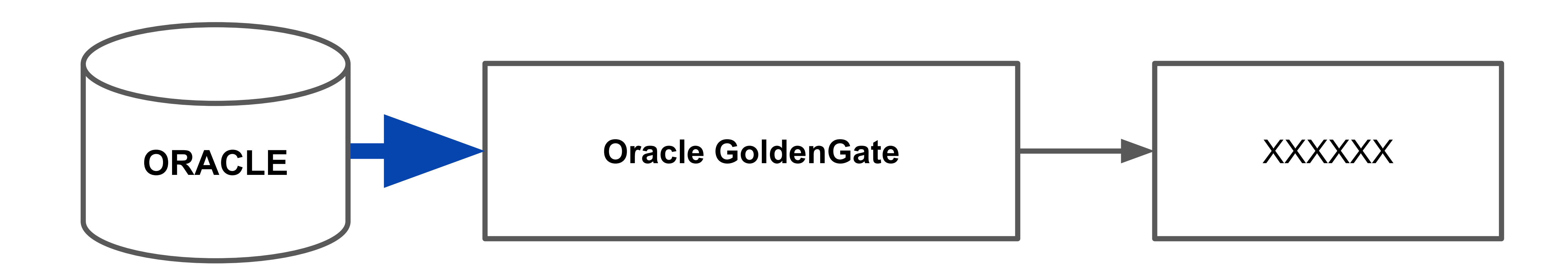


Почему так?



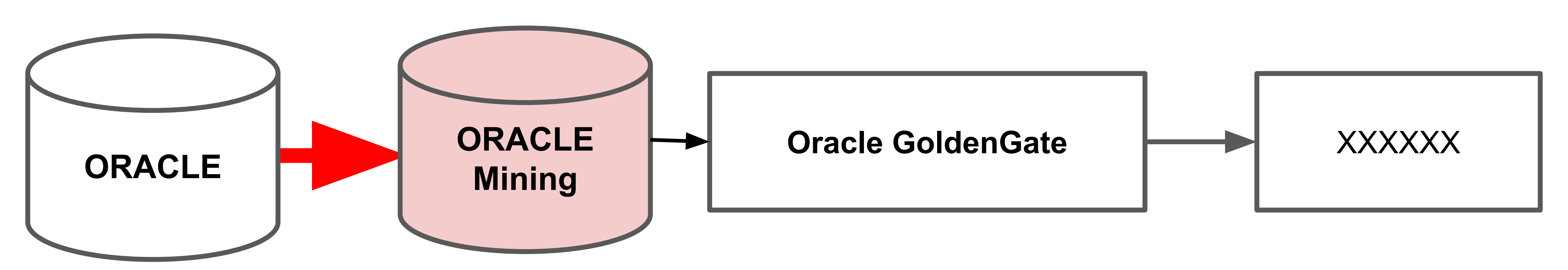


Почему так?





Почему так?


ORACLE
ORACLE
Mining
Oracle GoldenGate
XXXXXX


HL
HighLoad++
Весна 2021

Остаавание репликации по времени (секунды)

~18 СЕКУНД

20 s
15 s
10 s
5 s
0 ns

15:00 15:30 16:00 16:30 17:00 17:30 18:00 18:30 19:00 19:30 20:00 20:30



Почему так?

ORACLE

ORACLE Mining

Oracle GoldenGate

XXXXXX

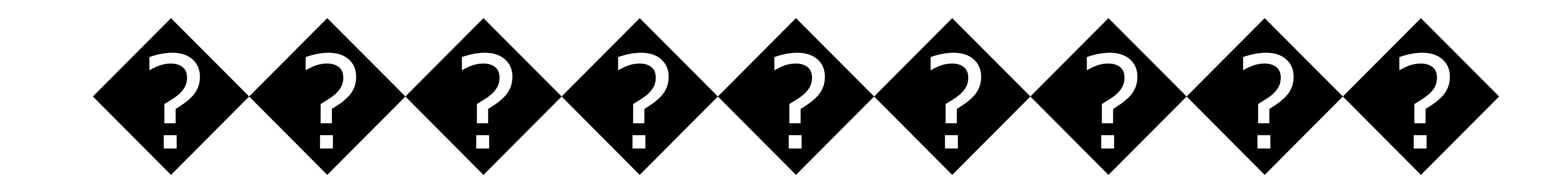
ORACLE



Итого

- Достигли реалтайма
- Mining работает как слой изоляции от сбоев
- Нужно работать с Redo-логами, не с Archive





CP1251? UTF-8?



XML

FORMATXML ENCODING UTF-8



XML

FORMATXML ENCODING UTF-8

……..

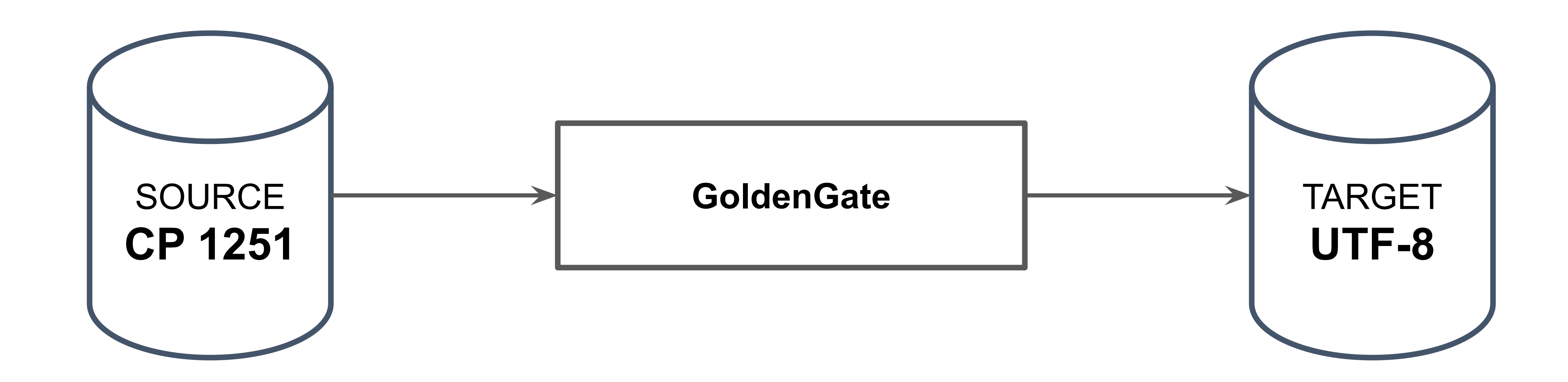
<columns>

<column name=”COLUMN” key=true>**НОРМАЛЬНЫЙ ТАКОЙ UTF**</column>

</columns>

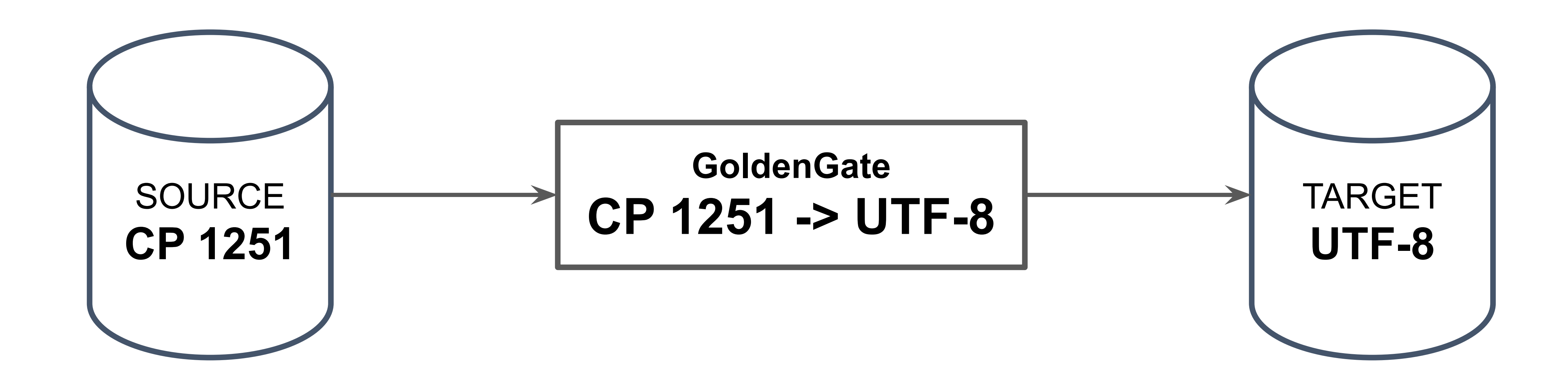


USEREXIT


SOURCE
CP 1251
GoldenGate
TARGET
UTF-8

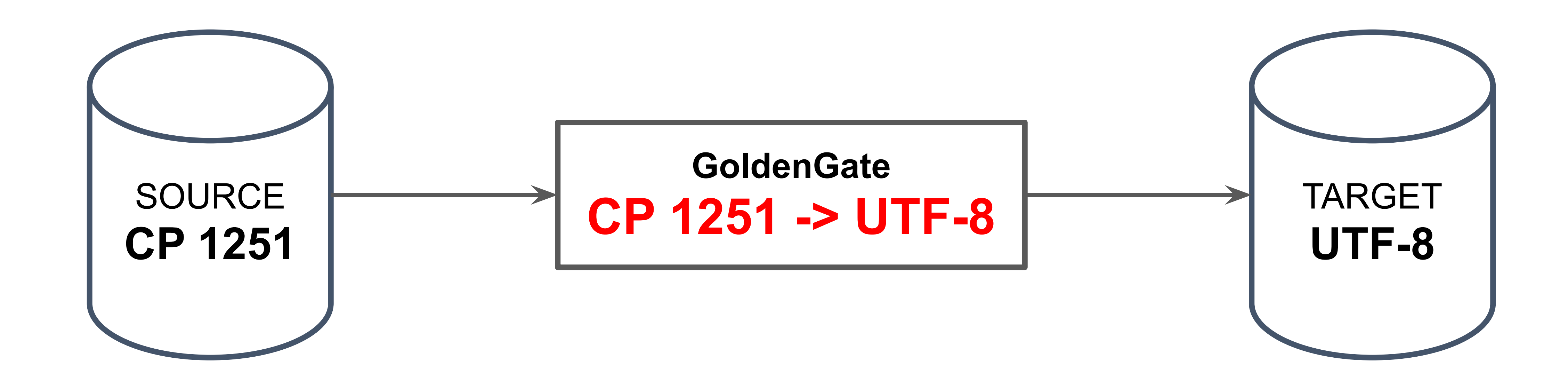

HL
HighLoad++
Весна 2021

USEREXIT


SOURCE
CP 1251
GoldenGate
CP 1251 -> UTF-8
TARGET
UTF-8

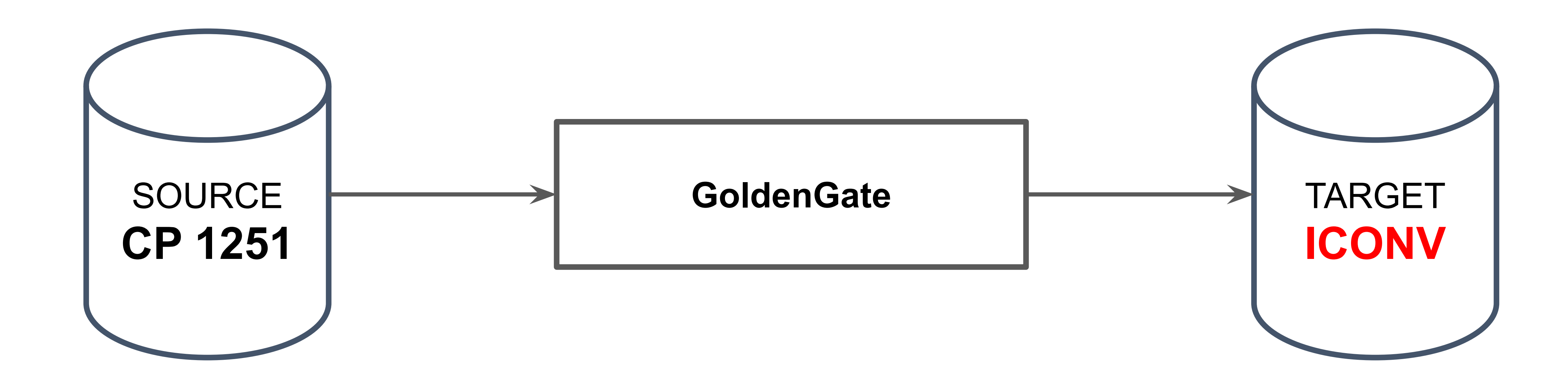

HL
HighLoad++
Весна 2021

USEREXIT


SOURCE
CP 1251
GoldenGate
CP 1251 -> UTF-8
TARGET
UTF-8


HL
HighLoad++
Весна 2021

iconv


SOURCE
CP 1251
GoldenGate
TARGET
ICONV


HL
HighLoad++
Весна 2021

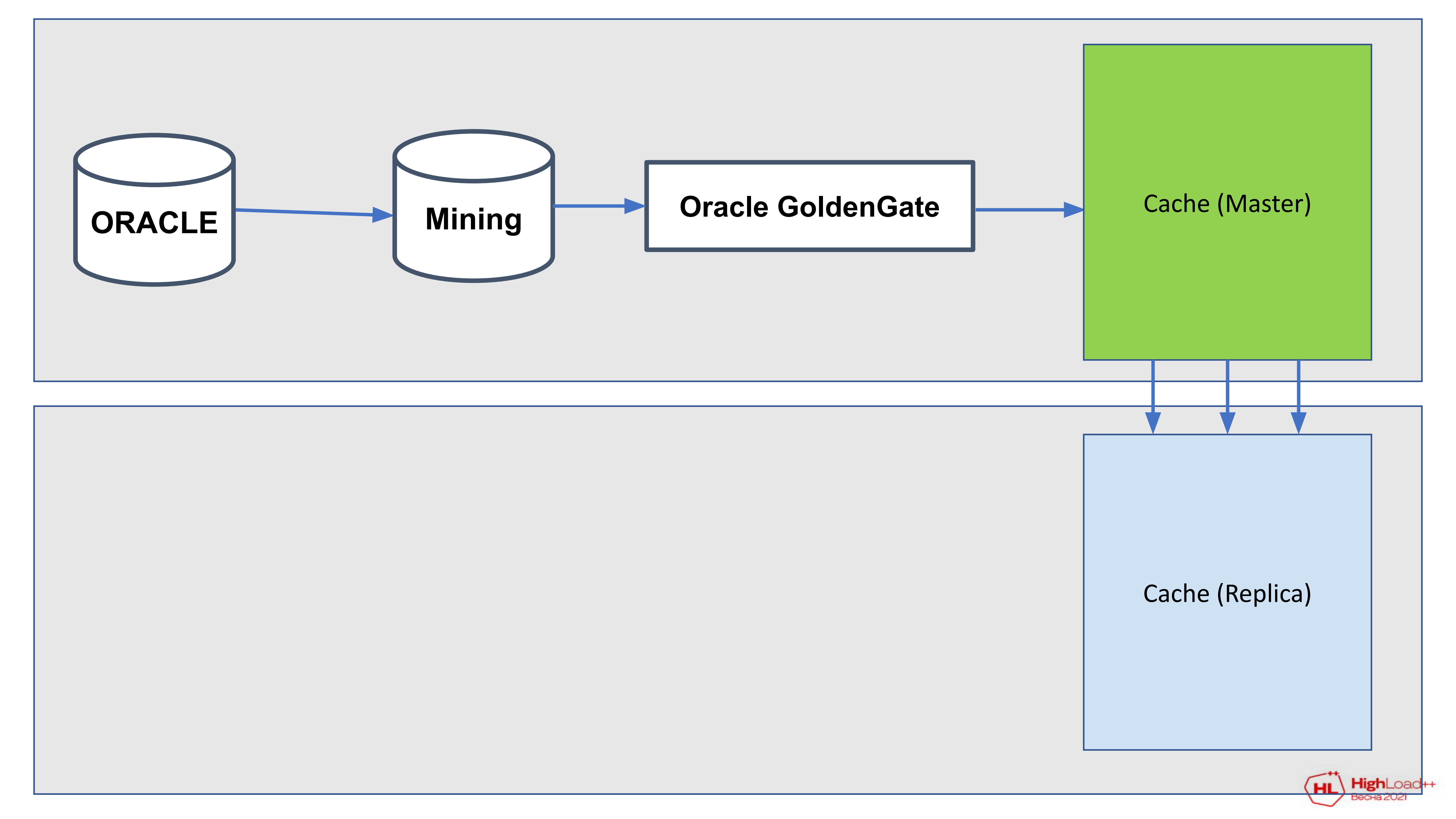
Итого

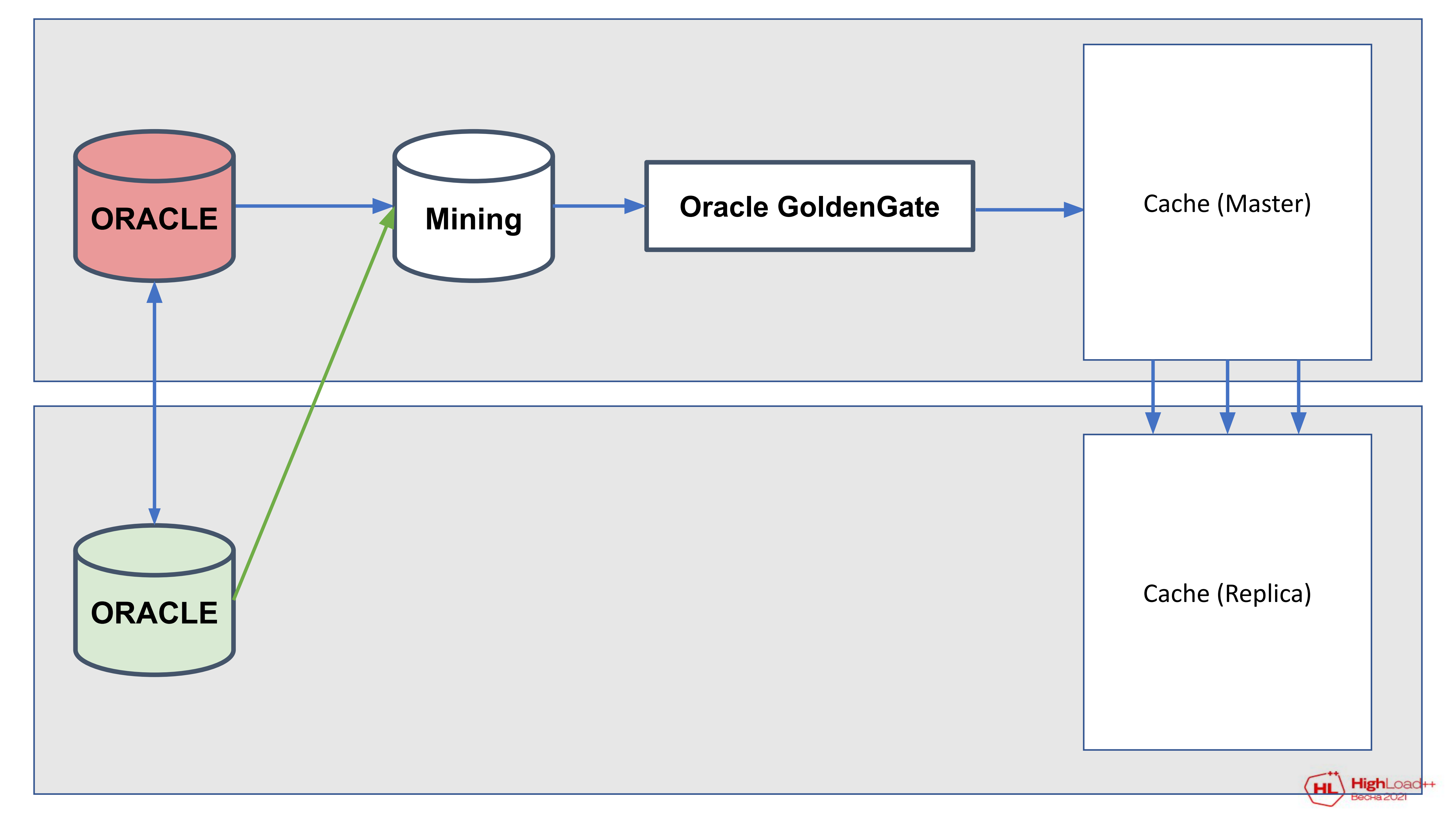
- XML — проще всего
- USEREXIT — конвертирует сам, но нам не покатило
- iconv — работает, но есть нюансы

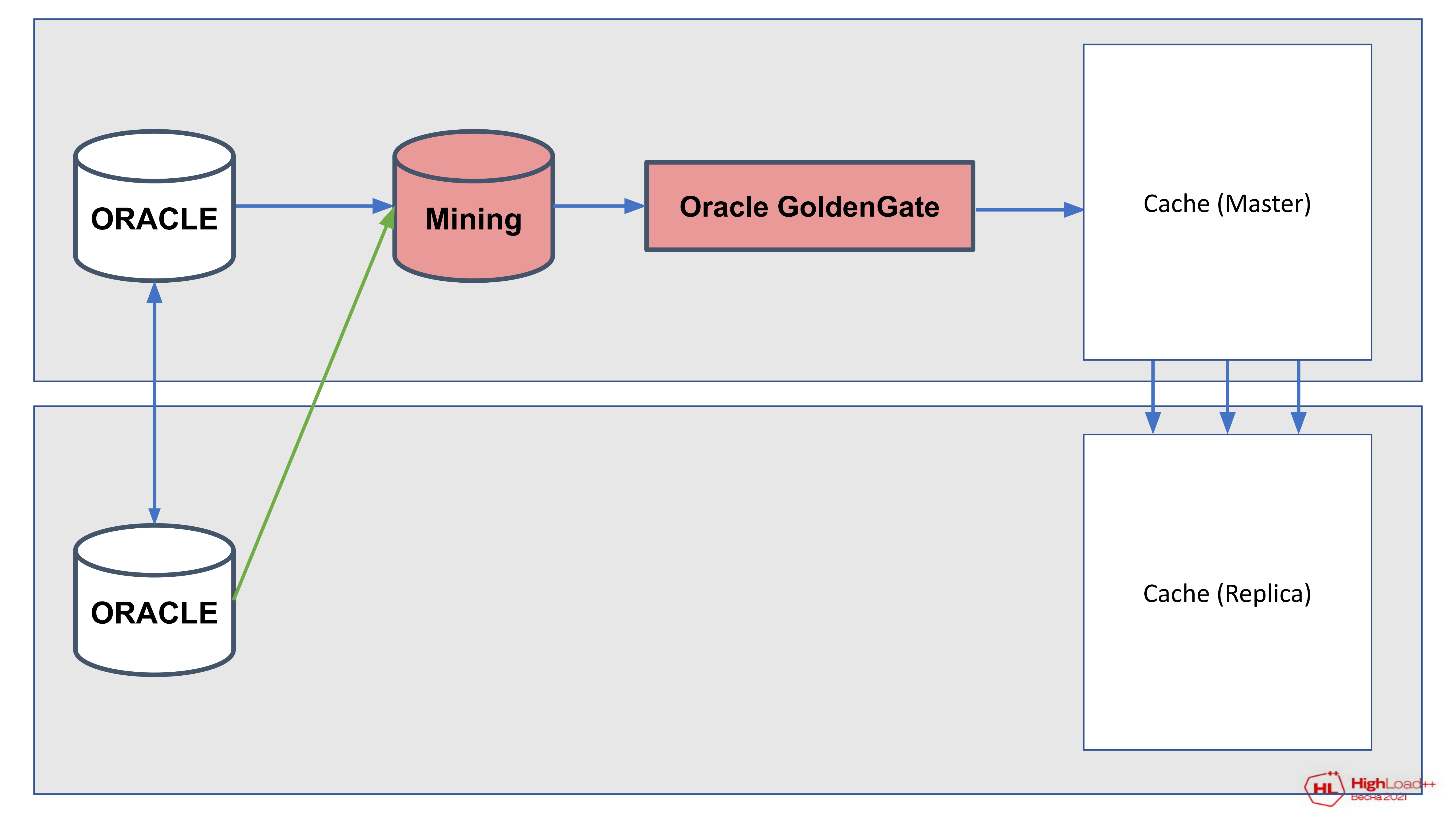


Failover

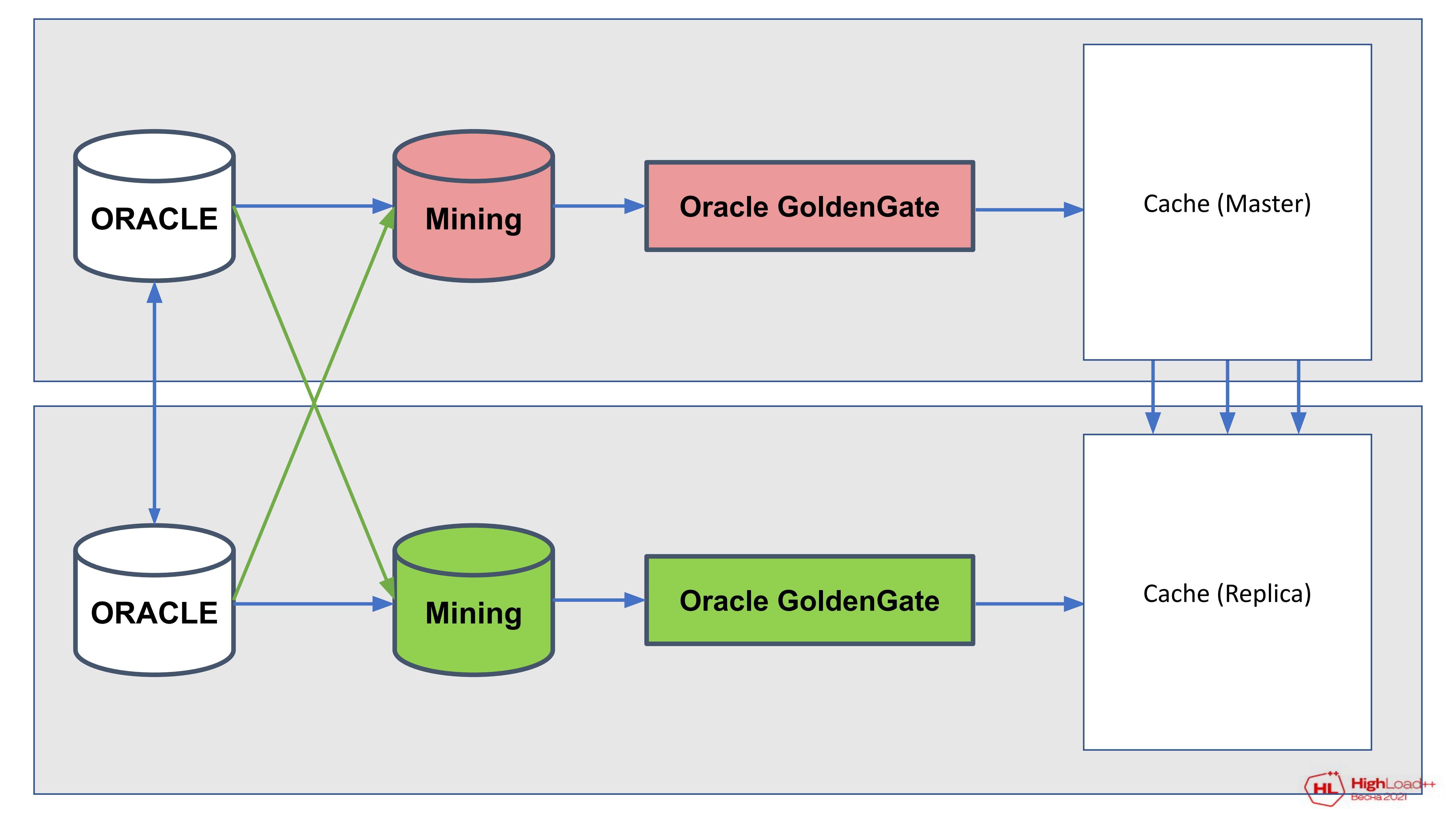



ORACLE
Mining
Oracle GoldenGate
Cache (Master)
Cache (Replica)
HL
HighLoad++
Весна 2021

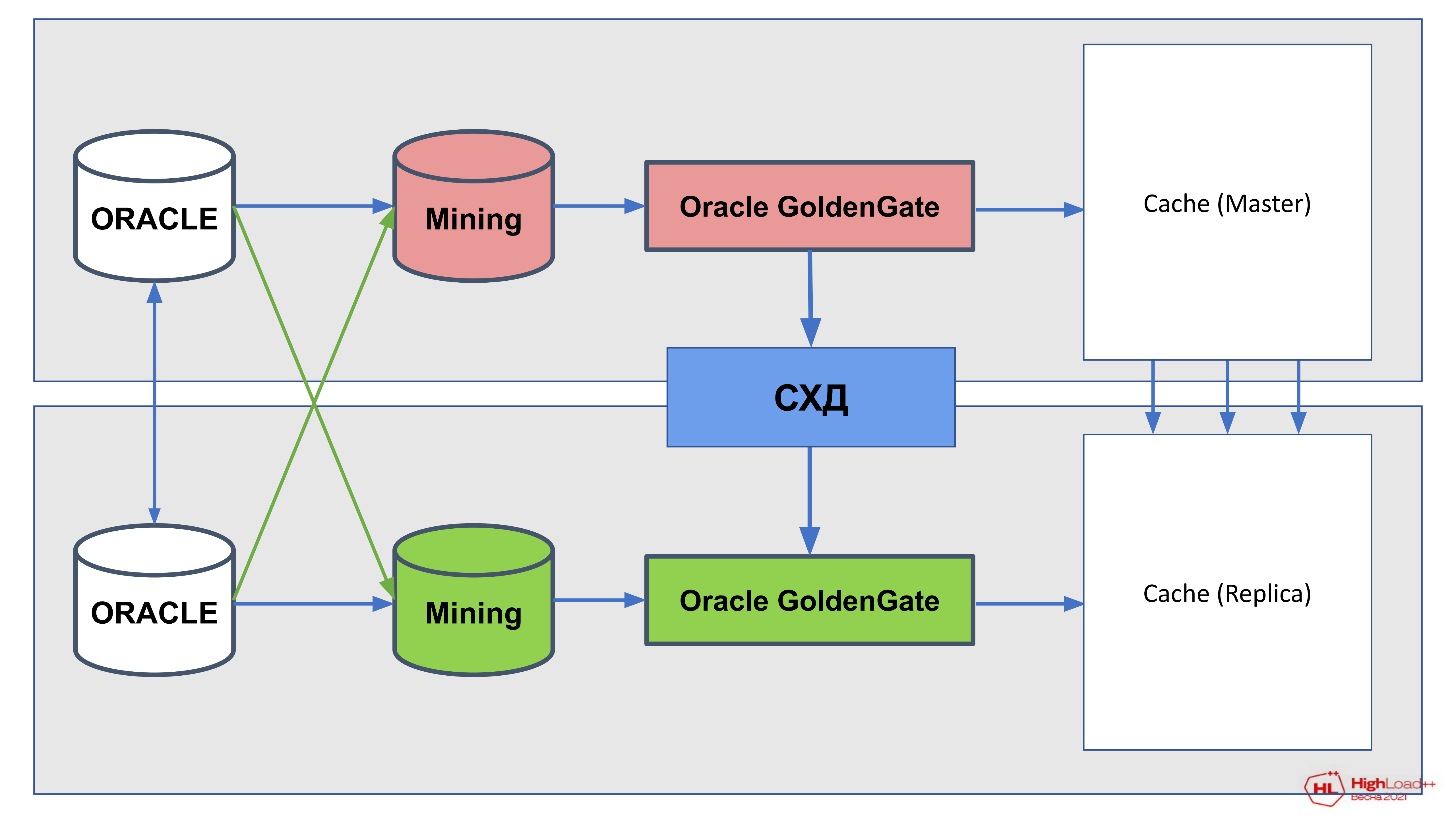

ORACLE
Mining
Oracle GoldenGate
Cache (Master)
ORACLE
Cache (Replica)
HL HighLoad++
Весна 2021



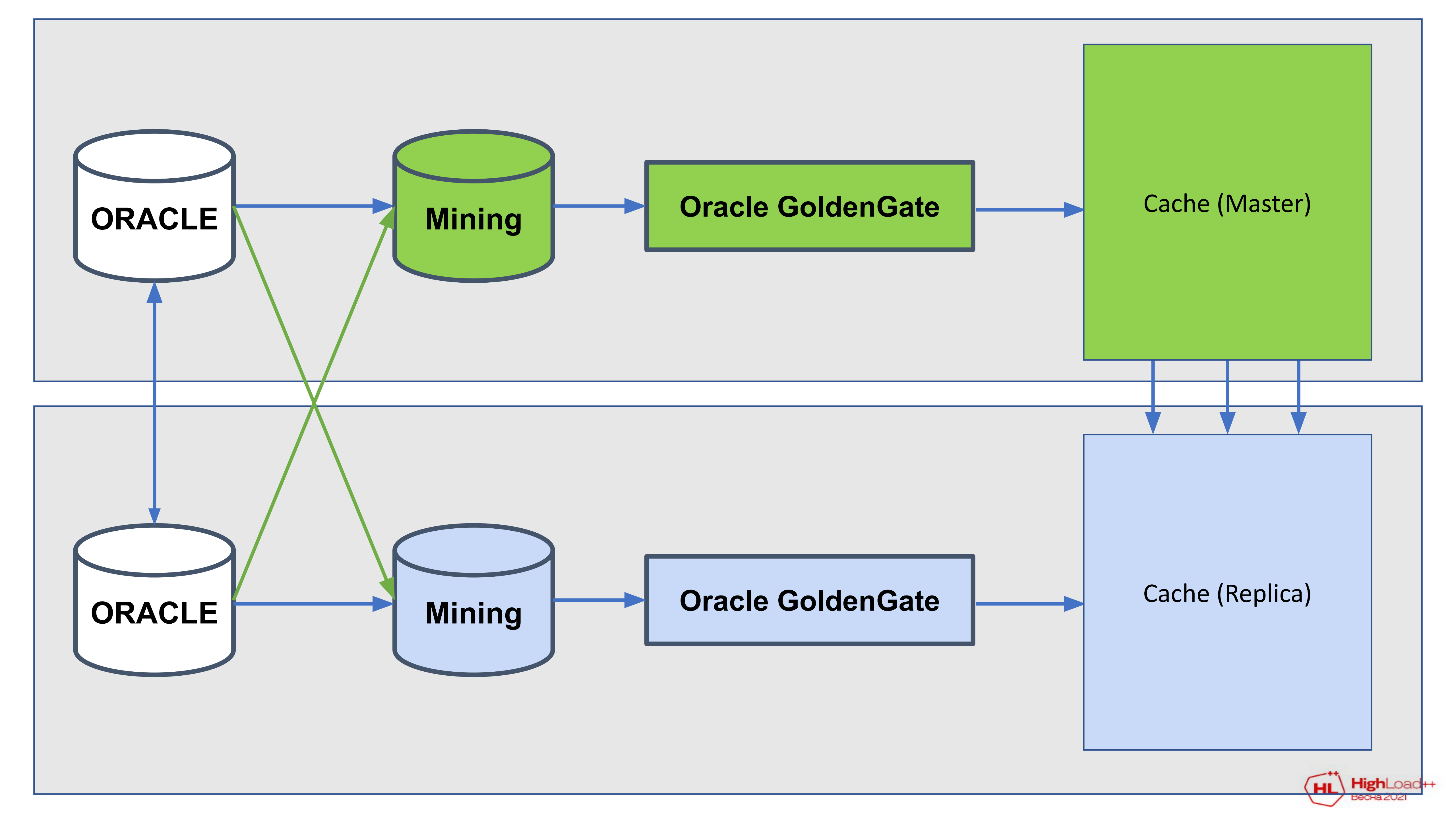
ORACLE
Mining
Oracle GoldenGate
Cache (Master)
ORACLE
Cache (Replica)
HL
HighLoad++
Весна 2021

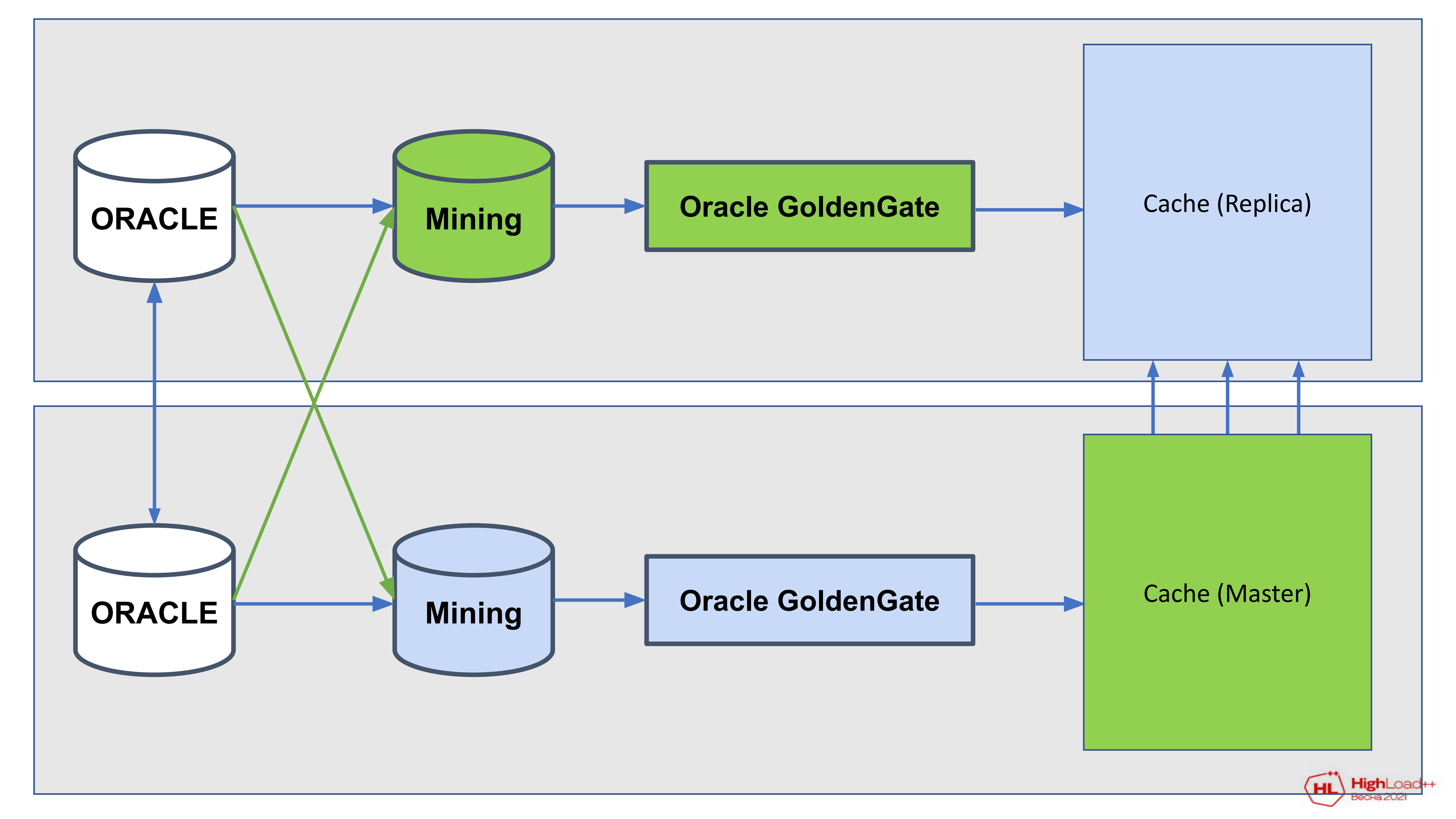


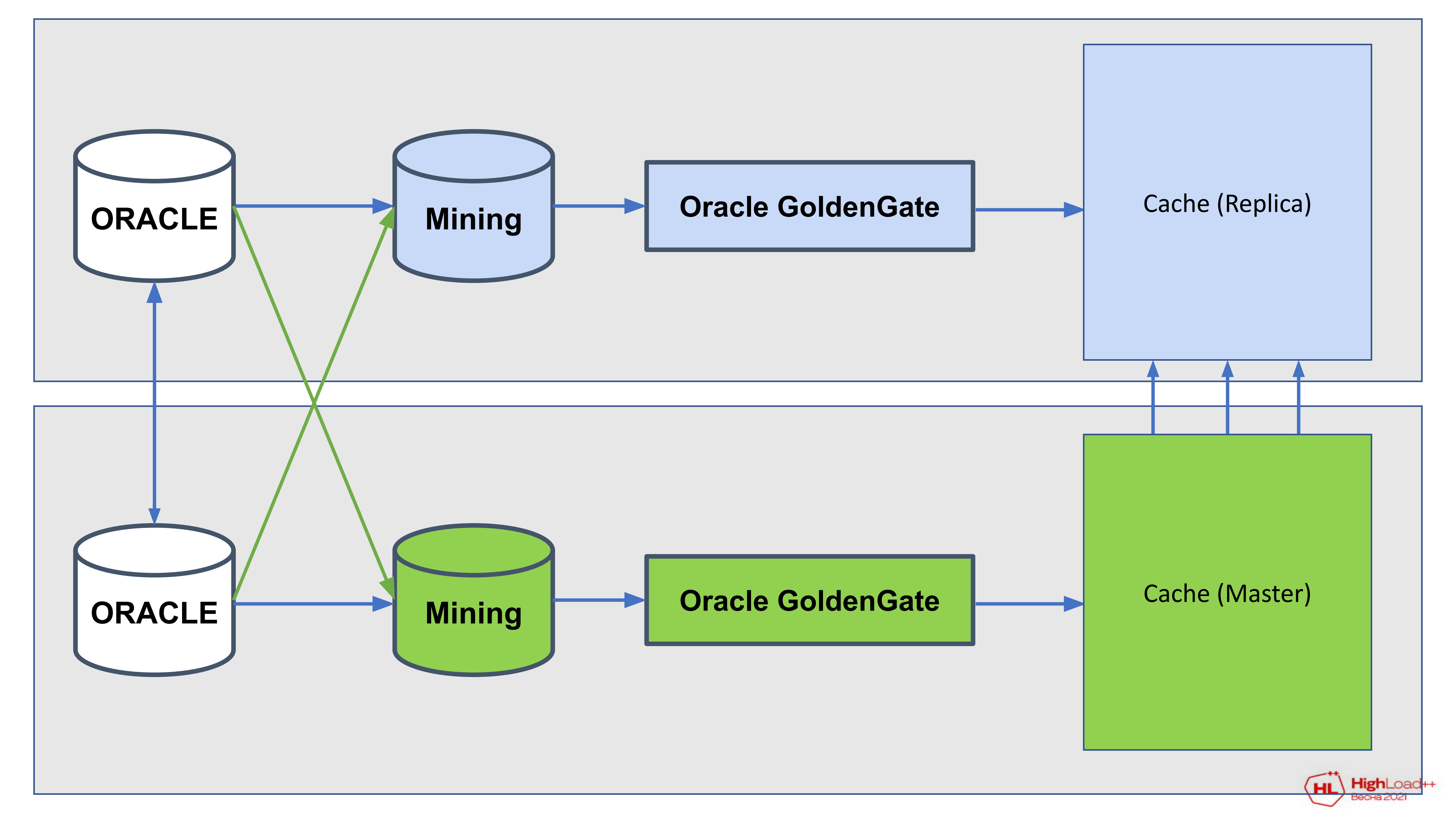
ORACLE
Mining
Oracle GoldenGate
Cache (Master)
ORACLE
Mining
Oracle GoldenGate
Cache (Replica)
HL HighLoad++
Весна 2021



ORACLE
Mining
Oracle GoldenGate
Cache (Master)
СХД
ORACLE
Mining
Oracle GoldenGate
Cache (Replica)
HL
HighLoad++
Весна 2021


ORACLE
Mining
Oracle GoldenGate
Cache (Master)
ORACLE
Mining
Oracle GoldenGate
Cache (Replica)
HL
HighLoad++
Весна 2021


ORACLE
Mining
Oracle GoldenGate
Cache (Replica)
ORACLE
Mining
Oracle GoldenGate
Cache (Master)
HL
HighLoad++
Весна 2021


ORACLE
Mining
Oracle GoldenGate
Cache (Replica)
ORACLE
Mining
Oracle GoldenGate
Cache (Master)
HL
HighLoad++
Весна 2021

Итог

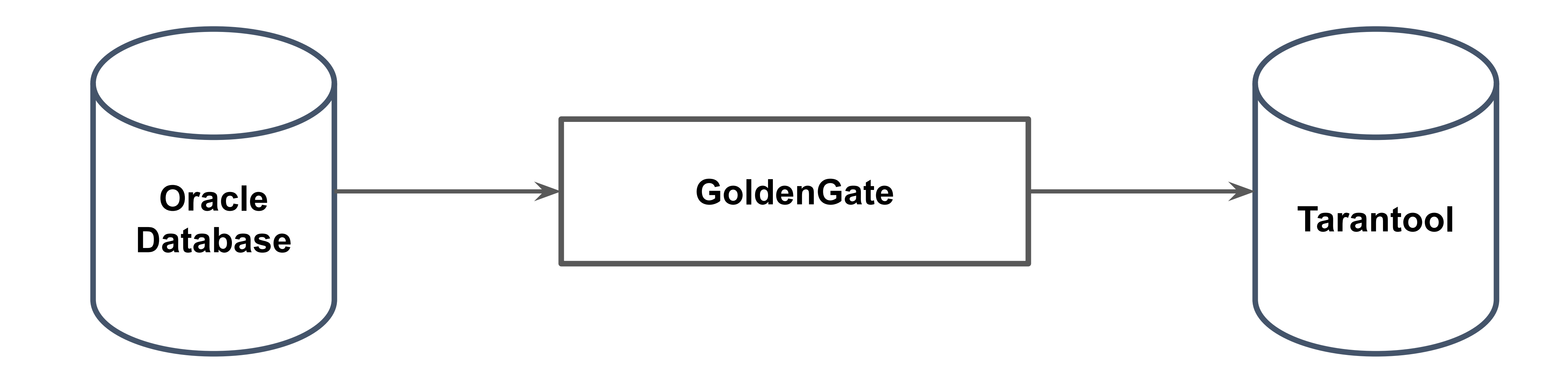
- время failover'a в рамках SLA
- фэйловер автоматический
- фэйловер GG относительно дешевый



Немного про эксплуатацию



Нет “обратной связи”





Нет “обратной связи”

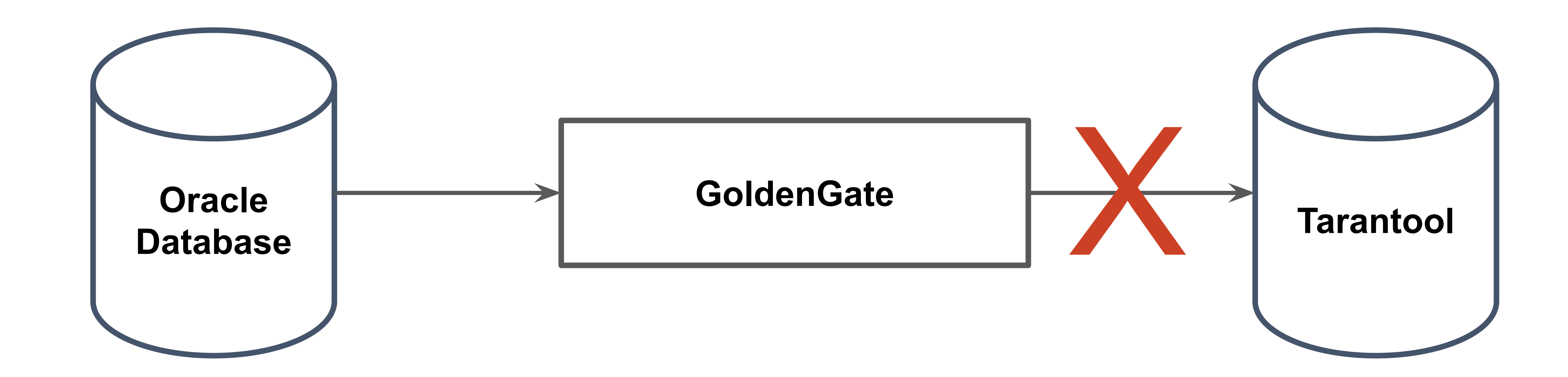
Oracle Database

GoldenGate

Tarantool

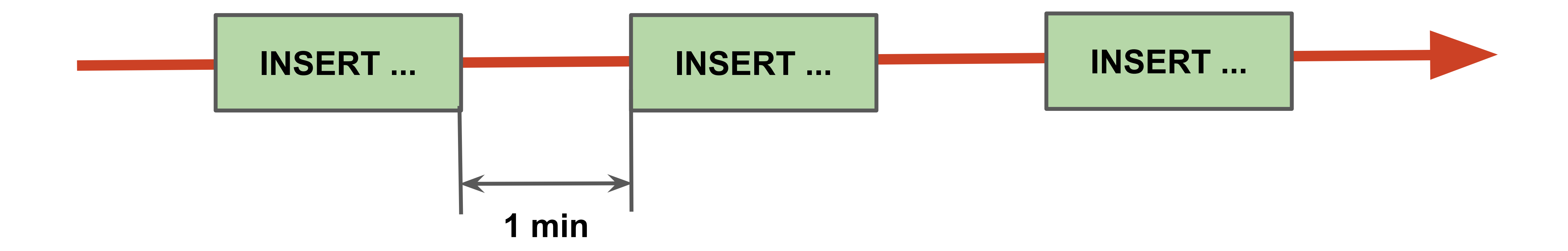


Нет “обратной связи”





Heartbeat





Мониторинг: лаг репликации end 2 end

```
<tokens>
    <token name="X_COMMIT_TIME">...</token>
</tokens>
```

LAG = NOW - X_COMMIT_TIME



Итог

- **Непрерывность репликации**
- **Лаг репликации**
- Много чего еще — статусы всех процессов, нагрузка на downstream базе, etc.



Итого по технике

- Характеристики источника
- Структура витрины
- XML — легальный и самый простой способ CDC
- USEREXIT — более технологичный способ CDC
- Важны эксплуатационные характеристики



Итог





Tarantool на Highload

